A024951

19 REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

18 1. REPORT NUMBER: AFGL-TR-76-0056

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER: 9

4. TITLE (and Subtitle): 6 COMPUTATIONAL TECHNIQUES FOR THE STUDY OF INFRASOUND PROPAGATION IN THE ATMOSPHERE,

5. TYPE OF REPORT & PERIOD COVERED: Final Report, 15 Oct 73 - 31 Dec 75,

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): 10

Allan D. Pierce
Wayne A. Kinney

8. CONTRACT OR GRANT NUMBER(s): 15 F19628-74-C-0065

9. PERFORMING ORGANIZATION NAME AND ADDRESS:

School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62101F, 17 76390102, 16 AF-7639

11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Geophysics Laboratory, Hanscom AFB, Massachusetts 01731, Monitor/Elisabeth F. Iliff/LWW

12. REPORT DATE: 11 13 Mar[redacted]76

13. NUMBER OF PAGES: 186

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 12 187p.

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Acoustics
Acoustic-gravity waves
Wave propagation
Intrasound
Atmospheric acoustics
Waveform synthesis
Guided waves
Nuclear explosion effects
Aerial acoustic waves

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A discussion is given of theoretical studies on infrasound propagation through the atmosphere which were carried out under the contract. Topics discussed include (1) the modification and adaptation of a computer program for the prediction of pressure signatures at large distances from nuclear explosions to include leaking guided modes, (2) the nature of guided over

infrasonic modes at higher infrasonic frequencies and the methods of extending waveform synthesis procedures to include higher frequencies, and (3) the propagation of infrasonic pressure pulses past the antipodes (over halfway around the globe). Summaries are included of all papers, theses, and reports written under the contract and conclusions and recommendations for future studies are given. An updated version of the computer program IFRASONIC WAVEFORMS originally given by Pierce and Posey in the report AFCRL-70-0134 is included as an appendix.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The authors wish to express their thanks to Mrs. E. F. Iliff of the Air Force Cambridge Research Laboratories for her interest and encouragement during the course of the research reported here.

Chapter I

INTRODUCTION

1.1 SCOPE OF THE REPORT

The present report summarizes investigations carried out by the authors during the years 1973-1976 on the propagation of low frequency pressure disturbances under Air Force Contract No. F19628-74-C-0065 with the Air Force Cambridge Research Laboratories, Bedford, Massachusetts. The study performed was theoretical in nature.

The central topic of this study was the generation and propagation of infrasonic waves in the atmosphere. The principal emphasis was on waves from man made nuclear explosions although certain aspects of the study pertain to waves generated by natural phenomena including, in particular, severe weather.

Specific topics considered during the study include the following:

1.) The adaptation of the computer program INFRASONIC WAVEFORMS to include leaking modes and to improve its accuracy in synthesizing early long period arrivals. (INFRASONIC WAVEFORMS is a digital computer program for the prediction of pressure signatures as would be detected at large horizontal distances following the detonation of a nuclear device in the atmosphere. The original version of this program was developed by Pierce and Posey[1] under a previous Air Force Contract [F19628-67-C-0217].) The developed theory for this adaptation has already been explained[2] in Scientific Report No. 1 of the present contract; the present report describes the numerical implementation of this theory (Chapter III), and gives some specific numerical examples. The complete current version of INFRASONIC WAVEFORMS is included here as Appendix A.

2.) The development of a ray acoustic model for the synthesis of higher frequency portions of infrasonic waveforms. The theory developed during

this study is given[3] in some detail in Scientific Report No. 2 and a discussion of this phase of the work is accordingly not repeated here.

3.) The modification of the multi-modal synthesis method to avoid truncation of upper limits on frequency integration. The method developed is presented here in Chapter IV and represents an extension of the W.K.B.J. technique to the case when the atmosphere has two sound channels. The resulting theory clarifies the problem of selection of modes for inclusion into the synthesis and leads to a relatively simple method for revising the synthesis program. (This revision, however, has not yet been carried out.)

4.) Study of infrasonic waveform synthesis for propagation near and past the antipodes. The method for doing this was briefly mentioned in the 1973 AFCRL report (pages 25 and 26) by Pierce, Moo, and Posey[4]. In Chapter V of the present report the theory underlying this is given and some numerical examples are given.

In Chapter II, we list all of the reports, papers, and theses which were written during the course of this study. The abstracts given there plus the abstract of the present report should be considered as a comprehensive summary of the accomplishments during the contracting period. In subsequent chapters of the present report, detailed discussions are given of some of the topics described above. In Chapter VI, some recommendations are made for future work in the field.

1.2 BACKGROUND OF THE REPORT

The general topics of infrasound propagation, generation, and detection have been of considerable interest to a large segment of the scientific community for some time. A published bibliography (the existence of which allows us to omit extensive citations here) lists [Thomas, Pierce, Flinn, and Craine, 1971][5] over 600 titles, most of which are directly concerned with infrasound. Literature pertaining to the infrasonic detection of nuclear explosions constitutes a considerable portion of these. Earlier work by Rayleigh [1890][6], Lamb [1908,1910][7], G. I. Taylor [1929,1936][8], Pekeris [1939,1938][9], and Scorer [1950][10],

among others, which was concerned with waves from the Krakatoa eruption [Symond, 1888][11] and from the great Siberian meteorite [Whipple, 1930][12] is also directly applicable to the understanding and interpretation of nuclear explosion waves.

The present report thus merely summarizes a continuation of a small number of facets of a lengthy pattern of research which has been carried on by a large number of investigators in the past. In a more restricted sense, the work reported here represents a continuation of work done in three previous studies performed under contract for Air Force Cambridge Research Laboratories. The first of these was Air Force Contract No. AF19(628)-3891 with Avco Corporation during 1964-1966; the second was Air Force Contract No. AF19628-67-C-0217 with the Massachusetts Institute of Technology during 1967-1969, the third was AF19628-70-C-0008 (also with M.I.T) during 1970-1972. Summaries of the earlier work may be found in the appropriate final reports by Pierce and Moo [1967][13], by Pierce and Posey [1970][1], and by Pierce, Moo, and Posey [1973][4].

One of the principal results of the first two aforementioned previous contracts was a computer program INFRASONIC WAVEFORMS; the deck listing of the then current version of which is given in the report by Pierce and Posey [1970][1]. This program enables one to compute the pressure waveform at a distant point following the detonation of a nuclear explosion in the atmosphere. The primary limitation on the program's applicability to realistic situations is that the atmosphere is assumed to be perfectly stratified. However, the temperature and wind profiles may be arbitrarily specified. The general theory underlying this program is somewhat similar to that developed by Harkrider [1964][14] but differs from his in that it incorporates background winds and in that it has a different source model for a nuclear explosion.

Chapter II

PAPERS, THESES AND REPORTS

The following gives author, title, and abstract of papers, theses, and reports written during the course of this project.

2.1 A. D. Pierce, "Theory of Infrasound Generated by Explosions," Colloque International sur les Infra-Sons, Proceedings (Centre National de la Recherche Scientifique (CNRS) 15, quai Anatole France, 75700 Paris, September, 1973).

A review is given of recent studies by the author and his colleagues on infrasound generation by explosions and the subsequent propagation through the atmosphere. These studies include (i) development of computer programs for the prediction of pressure signatures at large distances from nuclear explosions, (ii) development of an alternative approximate model for waveform synthesis based on Lamb's edge mode, (iii) development of a geometrical acoustics' theory incorporating nonlinear effects, dispersion, and wave distortion at caustics, and (iv) theoretical models for the mechanisms of wave generation by explosions. The basic theory is briefly outlined in each case and some of the more significant results are explained in terms of simplified physical models. Such results include the predicted dependence of far field waveforms on energy yield and burst height, suggested techniques for estimating energy yield from waveforms, and an explanation of amplitude anomalies in terms of focusing and defocusing of horizontal ray paths.

2.2 W. A. Kinney, C. Y. Kapper, and A. D. Pierce, "Acoustic Gravity Wave Propagation Post the Antipode," J. Acoust. Soc. Amer. 55, S75 (A) (1974).

The previous theoretical formulations and numerical computations of pressure waveforms (such as described by Harkrider, Pierce, and Posey, and others) apply only to atmospheric traveling waves which have traveled less than 1/2 the distance around the earth. In the

present paper, a technique resembling that previously introduced by Brune, Nafe, and Alsop [Bull. Seismol. Soc. Am. 51, 247-257 (1961)] for elastic surface waves on the earth is discussed and applied to the acoustic-gravity wave propagation past the antipode problem. The principal modification to the older theory is a shift in phase of $\pi/2$ to the Fourier transform of the wave after it has traveled over halfway round the globe from the source. The source of the wave is presumed to be a nuclear explosion of given energy E. Numerically synthesized waveforms of antipodal arrivals are exhibited and compared with those for direct arrivals. The necessary modifications to the Lambmode model theory of Pierce and Posey [Geophys. J. Roy. Astron. Soc. 26, 341-368 (1971)] are also described.

2.3 C. Y. Kapper, "Leaky Infrasonic Guided Waves in the Atmosphere," J. Acoust. Soc. Amer. 56, S2 (A) (1974).

Prior theoretical formulations and computational techniques for the prediction of pressure waveforms generated by large explosions in the atmosphere have considered only fully ducted modes. In the present paper, a technique for including weakly leaking guided modes in concert with fully ducted modes is developed. Modification of previous theory includes the extension of the boundary condition at the upper halfspace to include a complex horizontal wavenumber. The major alterations to the computer program Infrasonic Waveforms (as described in report by Pierce and Posey, 1970) incurred consist of the computation of the imaginary part of the newly incorporated complex wavenumber, extension of the normal-mode dispersion function to lower frequencies, and a second-order correction factor to the phase velocity.

2.4 W. A. Kinney, "Asymptotic High-Frequency Behavior of Guided Infrasonic Modes in the Atmosphere," J. Acoust. Soc. Amer. 56, S2 (A) (1974).

Refinement of previous theoretical formulations and numerical computations of pressure waveforms as applied to atmospheric traveling infrasonic waves could include a description of their asymptotic behavior at high frequencies. In the present paper, calculations based on the W.K.B.J. approximation and similar to those introduced by

Haskell [J. Appl. Phys. 22, 157-167 (1951)] are performed to describe the asymptotic behavior of infrasonic guided modes as generated by a nuclear explosion in the atmosphere. The results of these calculations are then matched onto numerical solutions which have been given by Harkrider, Pierce and Posey, and others. It is demonstrated that the use of these asymptotic formulas in conjunction with a computer program which synthesizes infrasonic pressure waveforms has enabled the elimination of problems associated with high-frequency truncation of numerical integration over frequency. In this way, small spurious high-frequency oscillations in the computer solutions have been avoided.

2.5 C. Y. Kapper, Computational Techniques in Infrasound Waveform Synthesis, M. S. Thesis, School of Mechanical Engineering, Georgia Institute of Technology (December, 1974).

This thesis is concerned with two major theoretical and programming modifications to the digital computer program INFRASONIC WAVEFORMS for the synthesization of acoustic-gravity pressure waveforms generated by large explosions in the atmosphere. The first modification involves the extension of the guided mode approximation for pressure waveforms in the atmosphere into leaking mode regions and a consequent search for the imaginary part of the complex horizontal wave number. Particular results include a plot of phase velocity versus angular frequency showing the extension of the normal mode dispersion function into a leaky mode region for a multilayer atmosphere and a report on the search for the imaginary part of the complex horizontal wave number of a leaky mode for a two layer atmosphere. The second modification involves the extension of the systhesis of acoustic-gravity pressure waveforms to distances beyond the antipode. A phase shift is noted for waves passing through the antipode and a comparison of pre and post antipodal waveforms is presented.

2.6 W. A. Kinney, A. D. Pierce, and C. Y. Kapper, "Atmospheric Acoustic Gravity Modes Near and Below Low Frequency Cutoff Imposed by Upper Boundary Conditions," J. Acoust. Soc. Amer. <u>58</u>, S1 (A) (1975).

Perturbation techniques are described for the computation of the imaginary part of the horizontal wavenumber (k_I) for modes of

propagation. Numerical studies were carried out for a model atmosphere terminated by a constant sound-speed (478 m/sec) half space above an altitude of 125 km. The GR_0 and GR_1 modes have lower-frequency cutoffs. It was found that for frequencies less than 0.0125 rad/sec, the GR_1 mode has complex phase velocity; k_I varying from near zero up to a maximum of 3×10^{-4} km^{-1} with analogous results for the GR_0 mode. There is an extremely small frequency gap for each mode for which no poles in the complex k plane corresponding to that mode exist. These mark the transition from undamped propagation to damped propagation. In the complete Fourier synthesis, branch line contributions compensate for the absence of poles in these gaps. Computational procedures are described which facilitate the inclusion of the low-frequency portions of these modes in the waveform sysnthesis.

2.7 A. D. Pierce, and W. A. Kinney, Atmospheric Acoustic Gravity Modes at Frequencies Near and Below Low Frequency Cutoff Imposed by Upper Boundary Conditions, Report AFCRL-TR-75-0639, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. (March, 1976).

Perturbation techniques are described for the computation of the imaginary part of the horizontal wavenumber (k_I) for modes of propagation. Numerical studies were carried out for a model atmosphere terminated by a constant sound-speed (478 m/sec) half space above an altitude of 125 km. The GR_0 and GR_1 modes have lower-frequency cutoffs. It was found that for frequencies less than 0.0125 rad/sec, the GR_1 mode has complex phase velocity; k_I varying from near zero up to a maximum of 3×10^{-4} km^{-1} with analogous results for the GR_0 mode. There is an extremely small frequency gap for each mode for which no poles in the complex k plane corresponding to that mode exist. These mark the transition from undamped propagation to damped propagation. In the complete Fourier synthesis, branch line contributions compensate for the absence of poles in these gaps. Computational procedures are described which facilitate the inclusion of the low-frequency portions of these modes in the waveform sysnthesis.

2.8 A. D. Pierce, and W. A. Kinney, Geometric Acoustics Techniques in Far Field Infrasonic Waveform Synthesis, Report AFGL-TR-76-0055, Air

Force Cambridge Research Laboratories, Hanscom AFB, Mass. (1976).

A ray acoustic computational model for the prediction of long range infrasound propagation in the atmosphere is described. A cubic spline technique is used to approximate the sound speed versus height profile when values of sound speed are input for discrete height intervals. Techniques for finding ray paths, travel times, ray turning points, and rays connecting source and receiver are described. A parameter characterizing the spreading of adjacent rays (or ray tube area) is defined and methods for its computation are given. A method of determining the number of times a given ray touches a caustic is also described. Formulas are given for the computation of acoustic amplitudes and waveforms which involve a superposition of contributions from individual rays connecting source and receiver and which incorporate phase shifts at caustics. The possibility of a receiver being in the proximity of a caustic is considered in some detail and distinction is made between cases where the receiver is on the illuminated or shadow sides of a caustic. It is shown that a knowledge of parameters characterizing two rays at a point in the vicinity of a caustic provides sufficient information concerning the caustic to allow one to give a relatively accurate description of the acoustic field in its vicinity. The resulting theory involves Airy functions and uses concepts extrapolated from a theory published in 1951 by Haskell. The net result is a detailed computational scheme which should accurately cover the contingency of the receiver being near a caustic in the calculation of amplitudes and waveforms. A number of FORTRAN subroutines illustrating the method are given in an appendix. Limitations of the theory and suggestions for future developments are also given.

Chapter III

NUMERICAL SYNTHESIS OF WAVEFORMS INCLUDING LEAKING MODES

3.1 INTRODUCTION

The computer program INFRASONIC WAVEFORMS has been modified to allow inclusion of the contribution at low frequencies from leaking modes (specifically the GR_0 and GR_1 modes) to numerically synthesized infrasonic pressure waveforms. The procedure incorporated in this modification involves a partly manual calculation of the imaginary and real parts of the horizontal wavenumber, k_I and k_R, respectively) as discussed in Scientific Report No. 1.[2] That calculation is outlined in more detail here. The numbers presented for illustration are appropriate to the case of observations at 15,000 km distance from a 50 megaton explosion, where the explosion is at 3 km altitude, and where the atmosphere is assumed to contain no winds. (This restriction is just for illustrative purposes, but is not a limitation on the method.)

3.2 CALCULATION OF COMPLEX WAVENUMBERS

The first step in the calculation is to obtain values for the phase velocities $v_n(\omega)$, $v_a(\omega)$, and $v_b(\omega)$ for the GR_0 and GR_1 modes, and to obtain values for the elements $R_{11}(\omega,v)$ and $R_{12}(\omega,v)$ of the transmission matrix [R]. These calculations should be done, in particular, for all frequencies extending below the mode's nominal lower cutoff frequency. As mentioned in the previous report[2] , R_{11} and R_{12} depend on the atmospheric properties only in the altitude range 0 to z_T (the bottom of the upper halfspace), and these are independent of what is assumed for the upper halfspace. Also, $v_n(\omega)$ is the phase velocity for a given (n-th) mode for values of ω greater than the lower cutoff frequency ω_L; here $v_a(\omega)$ and $v_b(\omega)$ are values of the phase velocity ω/k at which the functions

R_{11} and R_{12}, respectively, vanish. For a given mode, the values of v_a and v_b chosen are those from the curves $v_a(\omega)$ and $v_b(\omega)$ which lie the closest of all such curves to the curve $v_n(\omega)$ for $\omega>\omega_L$.

As regards the calculation of R_{11} and R_{12}, the computer program INFRASONIC WAVEFORMS may be used, only with an alternate version of the subroutine TABLE. A copy of subroutine TABLE with the appropriate modifications incorporated and indicated is given in Appendix B. A deck listing of all of the input data that is required to obtain R_{11} and R_{12}, and that is appropriate to the running example, follows in Fig. 1. Values for R_{11} and R_{12} need only be calculated for phase velocities between, say, 143 and 0.3318 km/sec, and for frequencies between 0.001 rad/sec (as close to zero as would seem necessary and corresponding to a period of 6,283 sec or 1.75 hr) and the value of ω_B for the upper halfspace (.0128 rad/sec in our numerical example). In the calculations reported here, the upper frequency was taken as .031 rad/sec in order to confirm the continuity of the dispersion curves. A sample portion of the printout of R_{11} and R_{12} corresponding to the model atmosphere of Fig. 2 is given in Fig. 3 . The same set of output from a computer run which lists the R_{11} and R_{12} also includes the $v_n(\omega)$ for the GR_0 and GR_1 modes.

Values of $v_a(\omega)$ and $v_b(\omega)$ for these modes are obtained by two successive runs of INFRASONIC WAVEFORMS using in sequence two modified versions of the subroutine NMDFN. These modifications are so minor that the deck listing is omitted and we describe here the nature of the modifications.

To obtain $v_a(\omega)$, one need only change the third from end executable FORTRAN statement of subroutine NMDFN from

```
FPP = RPP(1,1)*A(1,2) - RPP(1,2)*(GU + A(1,1))        (3.1)
```

to

```
FPP = RPP(1,1).                                       (3.2)
```

```
$NAM1 NSTART=1, NPRNT=1, NPNCH=-1, NCMPL=-1 $END
$NAM2 IMAX=24,
ZI=1.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,25.,30.,35.,40.,45.,55.,
    65.,75.,85.,95.,105.,115.,125.,
T=292.,288.,270.,260.,249.,236.,225.,215.,205.,198.,205.,215.,227.,
    237.,249.,265.,260.,240.,205.,185.,184.,200.,250.,400.,570.,
LANGLE=1,
WINDY=25*0.0,
WANGLE=25*0.0
$END
$NAM4
THETKD =35.,
V1 = 0.143, V2 = 0.3318,
ØM1 = 0.001,   ØM2 = 0.031,
NØMI = 30, NVPI = 80,
MAXMØD = 10
$END
$NAM1 NSTART=6, NPRNT=1, NPNCH=-1, NCMPL=-1  $END
```

Figure 1. Listing of input data required to generate tabulations of R_{11} and R_{12} versus phase velocity and angular frequency in the vicinity of the dispersion curves for the GR_0 and GR_1 modes.

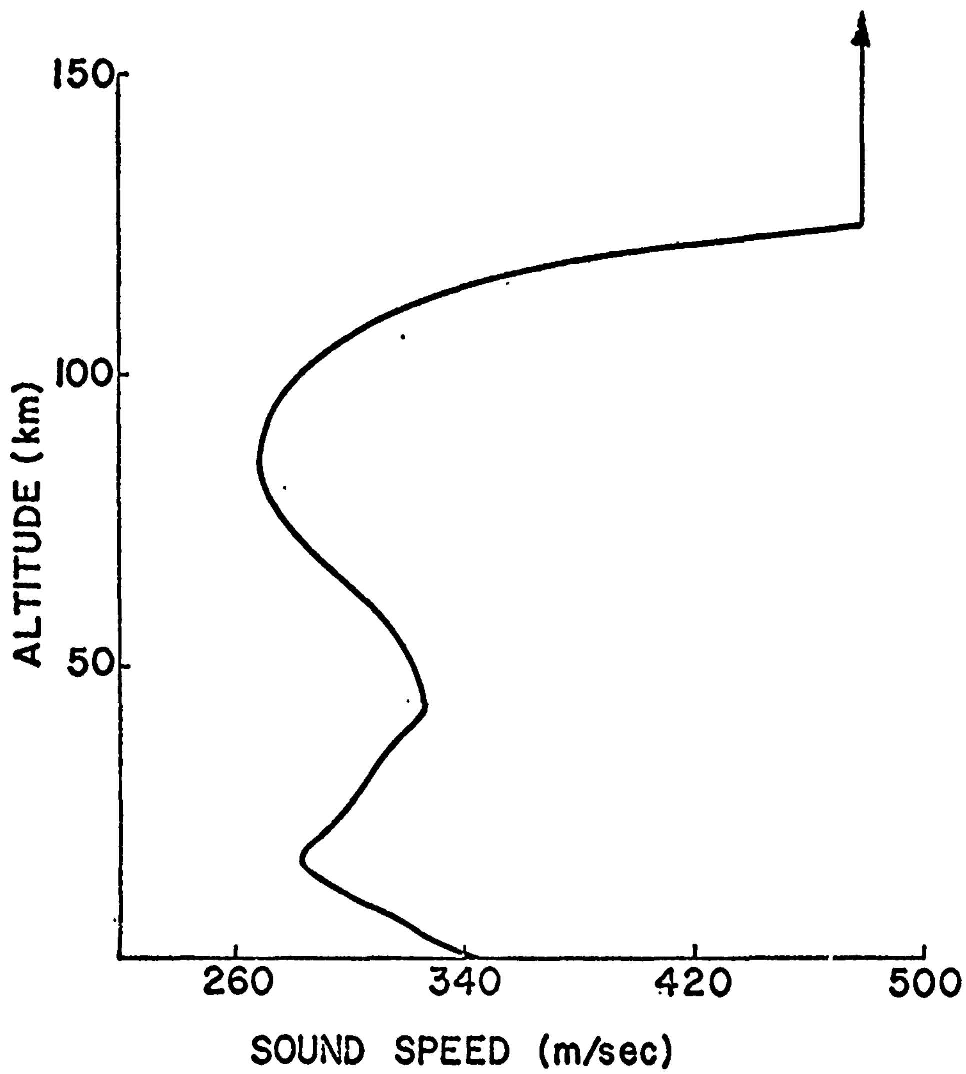


Figure 2. Model atmosphere showing sound speed versus altitude for numerical example treated in the present chapter. The atmosphere is bounded by an isothermal upper half space beginning at 125 km altitude.

v_p	R_{11}	R_{12}
OMEGA=	.30928-02	
.14300+00	.21671+01	-.65152+02
.14539+00	-.72963-01	-.22523+02
.14778+00	-.19992+01	.16898+02
.15017+00	-.34415+01	.49336+02
.15256+00	-.43200+01	.72532+02
.15495+00	-.46324+01	.85619+02
.15734+00	-.44356+01	.88883+02
.15973+00	-.38270+01	.83475+02
.16212+00	-.29260+01	.71114+02
.16451+00	-.18579+01	.53814+02
.16690+00	-.74204+00	.33657+02
.16929+00	.31761+00	.12611+02
.17168+00	.12376+01	-.75995+01
.17407+00	.19579+01	-.25568+02
.17646+00	.24418+01	-.40247+02
.17885+00	.26746+01	-.50952+02
.18124+00	.26605+01	-.57340+02
.18363+00	.24195+01	-.59371+02
.18602+00	.19834+01	-.57261+02
.18841+00	.13917+01	-.51424+02
.19080+00	.68860+00	-.42421+02
.19319+00	-.80574-01	-.30906+02
.19558+00	-.87165+00	-.17582+02
.19797+00	-.16447+01	-.31561+01
.20036+00	-.23637+01	.11690+02
.20275+00	-.29996+01	.26326+02
.20514+00	-.35295+01	.40198+02
.20753+00	-.39379+01	.52832+02
.20992+00	-.42158+01	.63849+02

Figure 3. Sample printout of R_{11} and R_{12} versus phase velocity for a fixed value of angular frequency. Output generated with the input data of Fig. 1.

To obtain $v_b(\omega)$, one need only change the same statement to

$$FPP = RPP(1,2). \tag{3.3}$$

The same limits for phase velocity and angular frequency as are used for the calculation of R_{11} and R_{12} should be used in the calculations for v_n, v_a, and v_b. In our example, when these limits are used, the GR_1 mode corresponds to mode #3, and the GR_0 mode corresponds to mode #4 for the case when $v_n(\omega)$ is calculated. For the cases when $v_a(\omega)$ and $v_b(\omega)$ are calculated, the GR_1 mode corresponds to mode #4 and the GR_0 mode corresponds to mode #6. A sample output listing of $v_n(\omega)$, $v_a(\omega)$ and $v_b(\omega)$ for the two modes is given in Fig. 4. An additional listing of $v_n(\omega)$, $v_a(\omega)$, and $v_b(\omega)$ for the two modes versus various values of ω is given in Table 1.

3.3 CALCULATION OF α AND β

The next step in the procedure is to manually calculate values for the variables α and β which enter into an approximate version [Eq. (9) in Scientific Report No. 1] of the eigenmode dispersion function. These parameters represent the partial derivatives of R_{11} and R_{12}, respectively, with respect to phase velocity v evaluated at $v=v_a$ and $v=v_b$, respectively. Since R_{11} and R_{12} also depend on ω, α and β may be considered as functions of angular frequency (but not of phase velocity).

It may be recalled that $v_a(\omega)$ and $v_b(\omega)$ are values for the phase velocity at which R_{11} and R_{12}, respectively, vanish. From the listing of, say, R_{11} versus v and ω, let the adjacent values R_{111}, R_{211}, R_{311} and R_{411} for R_{11} corresponding to the values for phase velocity v_{11}, v_{21}, v_{31} and v_{41}, respectively (for some chosen ω), such that v_{21} and v_{31} brackett a value for v_a; R_{211} and R_{311} would then be of opposite sign. In the listing of v, R_{11}, R_{12} for various ω, the values for v should all turn out to be equally spaced. Given this fact, it is possible to reasonably approximate α from the listings of R_{11} by the formula

$$\alpha = (1/\Delta v_1)([5/6]e_{11}+[1/12]f_{11}+[1/4]g_{11}h_{11}) \tag{3.4}$$

GR_0 MODE

ω	v_n	ω	v_a	ω	v_b
.012375	.31185608	.001030	.31205939	.001030	.31209836
.013407	.31181806	.002061	.31205552	.002061	.31209447
.014438	.31177597	.003093	.31204906	.003093	.31208799
.015469	.31172882	.004124	.31204001	.004124	.31207893
.016501	.31167509	.005156	.31202834	.005156	.31206727
.017532	.31161209	.006187	.31201405	.006187	.31205303
.018563	.31153394	.007218	.31199710	.007218	.31203620
.019070	.31148610	.008250	.31197748	.008250	.31201679
.019079	.31148516	.009281	.31195515	.009281	.31199478
.019595	.31142505	.010312	.31193006	.010312	.31197016
.019853	.31138841	.011344	.31190215	.011344	.31194291
.020111	.31134515	.012375	.31187139	.012375	.31191302
.020626	.31122480	.013407	.31183768	.013407	.31188045
.021658	.31029529	.014438	.31180093	.014438	.31184518
.021659	.31029116	.015469	.31176104	.015469	.31180714
.022005	.30790129	.016501	.31171786	.016501	.31176630
.022139	.30551142	.017532	.31167120	.017532	.31172258
.022173	.30475278	.018563	.31162087	.018563	.31167591
.022240	.30312155	.019595	.31156653	.019595	.31162620
.022329	.30073168	.020626	.31150781	.020626	.31157334
.022412	.29834181	.021658	.31144415	.021658	.31151721
.022490	.29595194	.022689	.31137478	.022689	.31145763
.022566	.29356207	.023720	.31129855	.023720	.31139444
.022639	.29117220	.024752	.31121368	.024752	.31132738
.022689	.28948366	.025783	.31111721	.025783	.31125619
.022710	.28878233	.026814	.31100382	.026814	.31118049
.022779	.28639246	.027846	.31086276	.027846	.31109984
.022846	.28400259	.028877	.31066848	.028877	.31101364
.022912	.28161272	.029909	.31034189	.029909	.31092114

GR_1 MODE

ω	v_n	ω	v_a	ω	v_b
.013407	.22781499	.001030	.24434330	.001030	.25073465
.013624	.22664568	.002061	.24409612	.001738	.25054440
.014040	.22425580	.003093	.24367787	.002061	.25042454
.014424	.22186593	.003655	.24337478	.003093	.24990029
.014438	.22177526	.004124	.24307887	.004124	.24915067
.014778	.21947606	.005156	.24228453	.005156	.24815906
.015107	.21708619	.006187	.24127431	.005160	.24815453
.015413	.21469631	.006445	.24098491	.006187	.24690257
.015469	.21423833	.007218	.24001984	.006963	.24576466
.015699	.21230644	.008181	.23859504	.007218	.24535036
.015966	.20991657	.008250	.23848240	.008250	.24346182
.016217	.20752670	.009281	.23660913	.008293	.24337478
.016453	.20513682	.009479	.23620517	.009281	.24118333
.016501	.20463309	.010312	.23432748	.009362	.24098491
.016675	.20274695	.010518	.23381529	.010260	.23859504
.016886	.20035708	.011344	.23153728	.010312	.23844346
.017085	.19796721	.011381	.23142542	.011034	.23620517
.017274	.19557733	.012115	.22903555	.011344	.23514877
.017454	.19318746	.012375	.22809942	.011712	.23381529
.017532	.19211887	.012752	.22664568	.012314	.23142542
.017626	.19079759	.013311	.22425580	.012375	.23116886
.017790	.18840772	.013407	.22381942	.012855	.22903555
.017946	.18601784	.013809	.22186593	.013345	.22664568
.018096	.18362797	.014255	.21947606	.013407	.22632580
.018240	.18123810	.014438	.21842295	.013790	.22425580
.018378	.17884823	.014659	.21708619	.014199	.22186593
.018510	.17645836	.015027	.21469631	.014438	.22036670
.018563	.17547997	.015364	.21230644	.014575	.21947606
.018638	.17406848	.015469	.21151653	.014922	.21708619

Figure 4. A sample output listing of $v_n(\omega)$, $v_a(\omega)$, and $v_b(\omega)$ for the GR_0 and GR_1 modes.

where

$$\Delta v_1 = v_{41} - v_{31} = v_{31} - v_{21} = v_{21} - v_{11} \tag{3.5a}$$

$$e_{11} = R_{311} - R_{211} \tag{3.5b}$$

$$f_{11} = R_{411} - R_{311} + R_{211} - R_{111} \tag{3.5c}$$

$$g_{11} = (R_{211} - R_{311})/e_{11} \tag{3.5d}$$

$$h_{11} = R_{311} + R_{211} - R_{111} - R_{411} \tag{3.5e}$$

In like manner, from the listing of R_{12} versus v and ω, if one lets the adjacent values R_{112}, R_{212}, R_{312}, and R_{412} for R_{12} correspond to the values for phase velocity v_{12}, v_{22}, v_{32}, and v_{42}, respectively (for some chosen ω), such that v_{22} and v_{32} bracket a value for v_b, then one can approximate β by the formula

$$\beta = (1/\Delta v_2)([5/6]e_{12} + [1/12]f_{12} + [1/4]g_{12}h_{12}) \tag{3.6}$$

where Δv_2, e_{12}, f_{12}, g_{12}, and h_{12} are defined by equations analogous to Eqs. (3.5) (last subscript changed from 1 to 2).

Because we use a numerical method (i.e., that described above) to calculate a derivative (it would be preferable to have an explicit formula), there is a small amount of numerical noise in the tabulation versus ω of α and β computed in the above manner. This noise is noticable only for the GR_1 mode and may for all practical purposes be filtered out by plotting α and β versus ω and then drawing smooth curves through the respective sets of points. (See Figs. 5 and 6.) While this procedure is somewhat laborious, it circumvents doing additional runs of the program to get values of R_{11} and R_{12} at more closely spaced values of phase velocity. It also circumvents a somewhat elaborate computer programming chore which would do

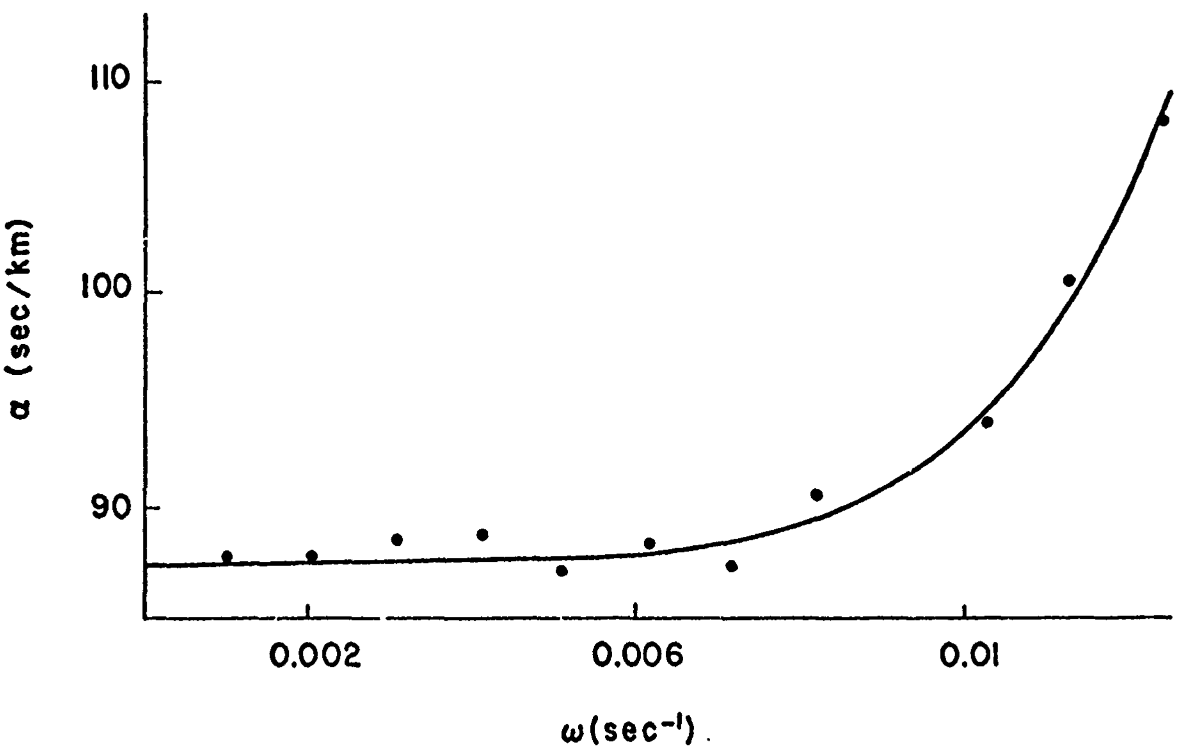


Figure 5. A plot of the parameter α versus ω for the GR_1 mode. The parameter α is $\partial R_{11}/\partial v_p$ evaluated at the phase velocity where $R_{11}=0$.

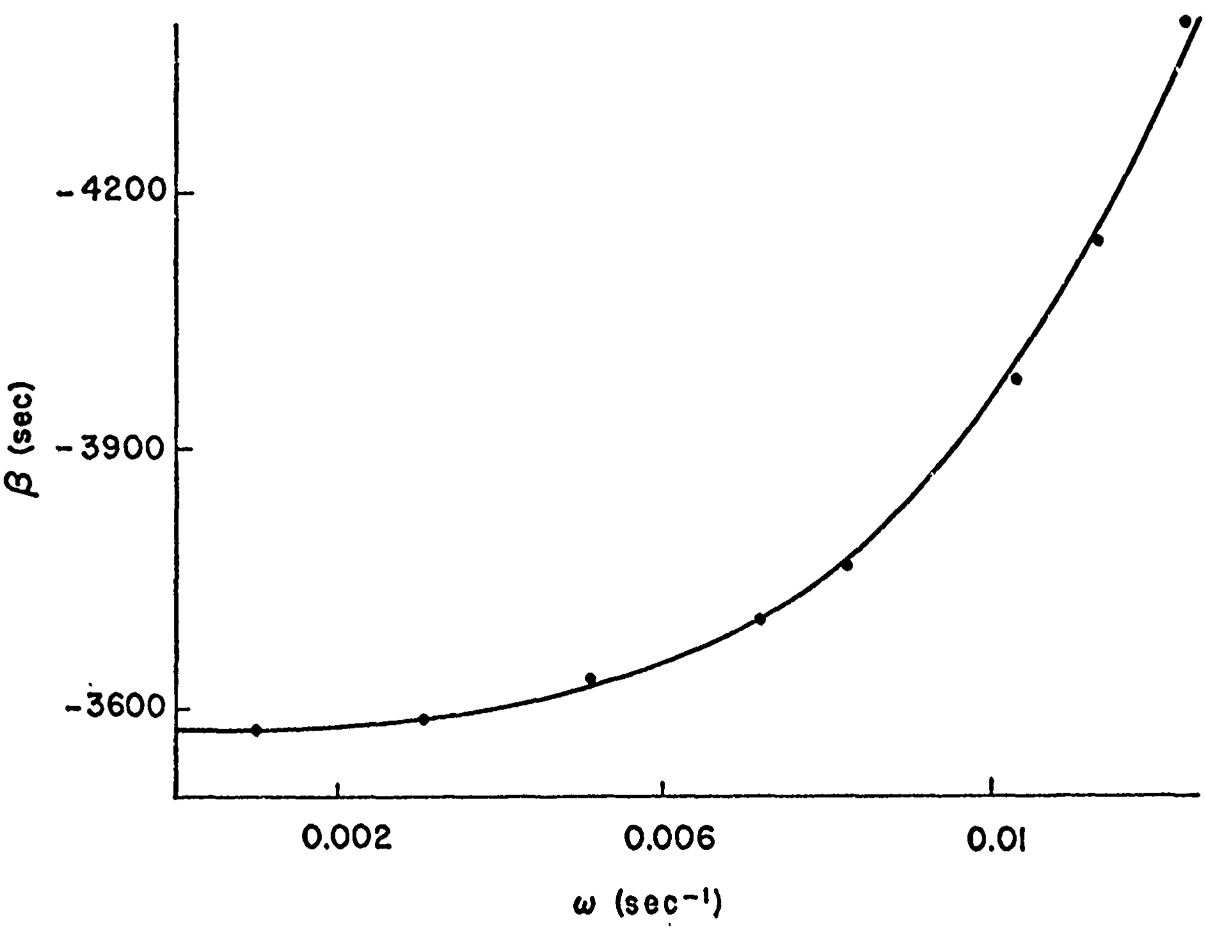


Figure 6. A plot of the parameter β versus ω for the GR_1 mode. The parameter β is $\partial R_{12}/\partial v_p$ evaluated at the phase velocity where $R_{12}=0$.

such steps automatically. (We suspect that the programming time would surpass all time which would ever actually be spent on manual circulations such as described above.) In any event, in view of the relatively small values of k_I which are actually obtained (as described further below) and in view of the recommendations (also given further below) concerning the use of the same k_I in many different types of calculations, the accuracy of the α and β so obtained is more than sufficient.

3.4 CALCULATION OF COMPLEX PHASE VELOCITY

The applicable expression for calculation of a mode's phase velocity (real above cutoff frequency, complex below) is Eq. (10a) in Scientific Report[2] No. 1 (which for brevity is not repeated here). This involves parameters v_a and v_b (whose computation is described in Sec. 3.1), and X, which may be considered as a function of ω and which is defined by Eq. (10b) in the prior report. This latter quantity X depends on β/α, A_{11}, G and A_{12}. The latter three are computed by taking the phase velocity as v_a and using Eqs. (4), (7a), and (7b) of the prior report. These calculations are straight forward, and do not require detailed explanation. Listings of G, A_{11}, A_{12}, and X for various values of ω and for the GR_1 and GR_0 modes are given in Table 1.

As explained in the prior report, below cutoff (that is, below ω_L = 0.0125 rad/sec for GR_1 and below ω_L = 0.0118 rad/sec for GR_0, in the running example) the real part k_R of the horizontal wave number is the real part of $\omega/v^{(1)}$, and the imaginary part k_I is the imaginary part of $\omega/v^{(1)}$. Finally, the extension by first iteration of the normal mode dispersion curves below cutoff is obtained by simply calculating ω/k_R. Listings of $v^{(1)}$, k_I, k_R, and ω/k_R for various ω for the GR_0 and GR_1 modes are given in Table 1. Plots of k_I and ω/k_R are given in Fig. 7.

3.5 INPUT DATA FOR GR_0 AND GR_1

The present version of INFRASONIC WAVEFORMS allows for the possibility of phase velocity ω/k_R, imaginary component k_I, and source free amplitude AMP to be input as functions of angular frequency ω for any given

GR_0 MODE

ω	v_a	v_b	α	β	A_{11}	A_{12}	G
0.001030	0.31205939	0.31209836	957.1	-2648.5	0.07064925	-1.3492340	0.02861746i
0.005156	0.31202834	0.31206727	917.4	-2783.7	0.07066928	-1.3497015	0.02585957i
0.008250	0.31197748	0.31201679	854.9	-2988.2	0.07070210	-1.3504677	0.02059949i
0.011344	0.31190215	0.31194291	767.9	-3254.2	0.07075075	-1.3515959	8.16470×10^{-3}i

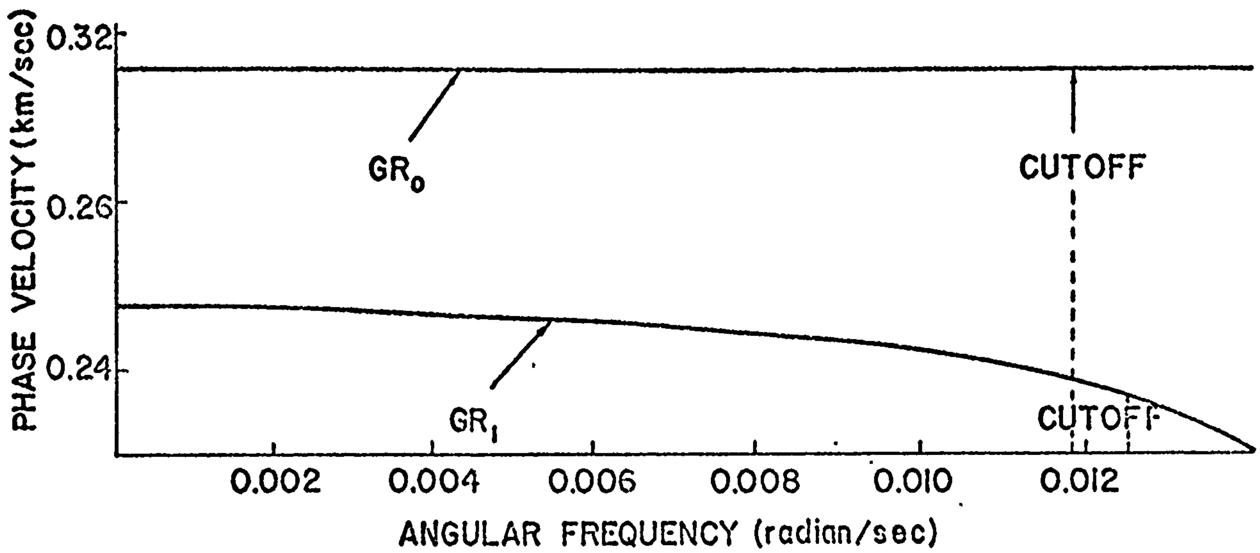
ω	X	k_I	k_R	ω/k_R
0.001030	0.14489848 + 0.05869314i	3.29323×10^{-8}	3.3007×10^{-3}	0.31205300
0.005156	0.15887128 + 0.05813477i	1.68605×10^{-7}	0.0165355	0.31202121
0.008250	0.18298964 + 0.05331514i	2.65003×10^{-7}	0.0264444	0.31197553
0.011344	0.22182228 + 0.02559851i	2.00717×10^{-7}	0.0363822	0.31189059

GR_1 MODE

ω	v_a	v_b	α	β	A_{11}	A_{12}	G
0.001030	0.24434330	0.25073465	87.4	-3578	0.13415774	-2.8317742	0.04359249i
0.005156	0.24284530	0.24815908	87.8	-3633	0.13695917	-2.8971705	0.04030849i
0.008250	0.23848240	0.24346182	89.6	-3770	0.14232483	-3.0224265	0.03397304i
0.011344	0.23153728	0.23514877	100.0	-4144	0.15281704	-3.2673565	0.01988061i

ω	X	k_I	k_R	ω/k_R
0.001030	1.9394832 + 0.63020518i	4.96794×10^{-5}	4.0319×10^{-3}	0.25546528
0.005156	1.9560589 + 0.57569611i	2.19268×10^{-4}	0.0204383	0.25269766
0.008250	1.9813366 + 0.47294644i	2.67086×10^{-4}	0.0335205	0.24759561
0.011344	1.9381840 + 0.25214654i	2.05014×10^{-4}	0.0474121	0.23926355

Table 1. Tabulation of frequency dependent parameters for the GR_0 and GR_1 modes. Tabulation is for frequencies below cutoff; definitions of the various quantities are given in the text and in Scientific Report No. 1.



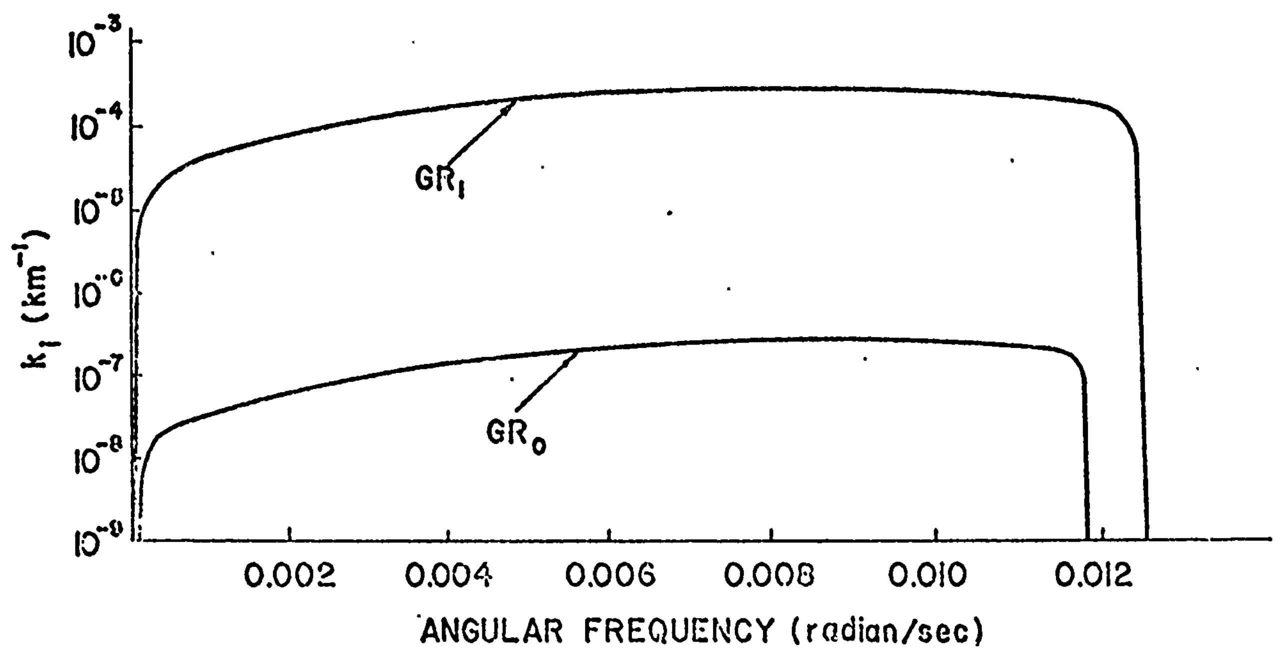


Figure 7. Numerically derived plots of phase velocity ω/k_R and of the imaginary part k_I of the complex horizontal wavenumber k versus angular frequency ω for the GR_0 and GR_1 modes. Nominal lower frequency cutoffs for these modes are as indicated. Note that k_I is identically zero above the cutoff frequency.

mode. The only modes for which this is necessary are GR_0 and GR_1. This input data is partly obtained by the procedure described above. Here we describe how the remaining portion of the input data is obtained.

To obtain values of phase velocity and source free amplitude at frequencies above cutoff one uses the current version of INFRASONIC WAVEFORMS with the variable NCMPL of NAMELIST NAM51 set less than zero. This gives an output essentially identical to what would be obtained with the original version of the program. The input data for this run would be the same as if one were computing waveforms without consideration of leaky modes. A sample listing of such input data is given in Fig. 8. The run will give mode numbers and tabulations of phase velocity VPHSE and amplitude AMP versus angular frequency OMEGA for the GR_0 and GR_1 modes at frequencies above cutoff. The only output which need be retained for future use are the tabulations of VPHSE versus OMEGA for these two modes, since amplitudes at frequencies above cutoff are computed automatically in the run which utilizes this information as input data. A sample tabulation of the pertinent output (for the running example considered here) is given in Fig. 9.

Input data of phase velocity VPHSE and amplitude AMP for frequencies below cutoff are obtained by a second run of the program, again with NCMPL < 0, only with the original model atmosphere replaced by one which has a thick intermediate layer plus on upper half space replacing the original upper half space. Thus, in the NAM2 input list, IMAX is increased by one, the original ZI and T are unchanged, but one adds a ZI for the new value of IMAX which is, say 100 km larger than the largest ZI for the original model atmosphere; the temperature T for the new IMAX + 1 layer (i.e. for the new upper half space) is set equal to an arbitrarily very large value (say, 2×10^7 °K). Doing this will artificially shift the cutoff frequencies for GR_0 and GR_1 down to values which are, for all practical purposes, equal to zero. The input data for this run should include choices of angular frequency and phase velocity limits (V1, V2, OM1, and OM2 of NAM4) which are appropriate for an exploration of the properties of GR_0 and GR_1 at frequencies below their original cutoff frequencies. It is imperative that OM2 not be too large since INFRASONIC WAVEFORMS will

```
$NAM1 NSTART=1, NPRNT=1, NPNCH=-1, NCMPL=-1  $END
$NAM2 IMAX=24,
ZI=1.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,25.,30.,35.,40.,45.,55.,
   65.,75.,85.,95.,105.,115.,125.,
T=292.,288.,270.,260.,249.,236.,225.,215.,205.,198.,205.,215.,217.,
   237.,249.,265.,260.,240.,205.,185.,184.,200.,250.,400.,570.,
LANGLE = 1,
WINDY = 25*0.0,
WANGLE = 25*0.0
$END
$NAM4
THETKD = 35.,
V1 = 0.15, V2 = 0.495,
ØM1 = 0.005, ØM2 = 0.1,
NOMI = 30, NVPI = 30,
MAXMOD = 8
$END
$NAM6  ZSCRCE = 3.0, ZØBS = 0.0  $END
$NAM8  YIELD = 50.E3  $END
$NAM10  RØBS = 15000.,
TFIRST = 46.2E3, TEND = 52.2E3,
DELTT = 15.,
IØPT = 11,
$END
$NAM1 NSTART=6 $END
```

Figure 8. Input data to obtain phase velocity versus angular frequency above cutoff frequency for the GR_0 and GR_1 modes.

GR_0 MODE

OMEGA	V_n
.01482759	.31175883
.01640552	.31167707
.01723448	.31162830
.01810345	.31157130
.01892241	.31150395
.01933190	.31145750
.01974138	.31140492
.02137931	.31079310
.02151039	.31060345
.02178079	.30980225
.02202362	.30762931
.02210359	.30614224
.02214436	.30539871
.02216121	.30502694
.02217751	.30465517
.02219828	.30416532
.02220876	.30391164
.02223357	.30316810
.02229504	.30168103
.02239972	.29870690
.02259155	.29275862
.02293273	.28086207
.02301724	.27771666
.02324256	.26896552
.02353065	.25706897
.02380369	.24517241
.02405701	.23327586
.02432538	.22137931
.02458369	.20948276
.02465517	.20622217
.02484741	.19758621
.02498335	.19163793
.02512335	.18568966
.02526862	.17974138
.02542062	.17379310
.02558111	.16784483
.02560520	.16487069
.02575227	.16189655
.02593679	.15594828
.02613807	.15000000

GR_1 MODE

OMEGA	V_n
.01482759	.21913010
.01601253	.20948276
.01640552	.20530285
.01711598	.19758621
.01723448	.19544661
.01756050	.19163793
.01795598	.18568966
.01810345	.18350434
.01832869	.17974138
.01865292	.17379310
.01892241	.16844746
.01895156	.16784483
.01909212	.16487069
.01922762	.16189655
.01933190	.15953747
.01948594	.15594828
.01973352	.15000000

Figure 9. Sample output of phase velocity versus angular frequency at frequencies above cutoff for the GR_0 and GR_1 modes corresponding to the input data of Fig. 8.

encounter numerical difficulties at higher frequencies when the height of the upper halfspace is as high as considered here. (If it were not for this fact, this run could be used to generate essentially the same information as in the previous run.) For comparison, Fig. 10 indicates the types of atmospheric profiles used in the two runs with NCMPL < 0.

The second run gives values for the source free amplitudes AMP and phase velocities VPHSE for the GR_0 and GR_1 modes for frequencies below cutoff. The latter of these are expected to be virtually identical to the ω/k_R which are obtained by the method described in Sec. 3.4. Also, the source free amplitudes are expected to match on smoothly to those obtained from the prior run for high frequencies even though the two model atmospheres are not identically the same. (This is because the energy transported by the GR_0 and GR_1 modes is predominantly in the lower atmosphere.) Furthermore, we expect these amplitudes to be virtually the same as would be obtained by the modified residue method described in Scientific Report No. 1 for the original model atmosphere. The actual amplitudes should have a small imaginary part, but in view of the relatively small values of the k_I (less than 10^{-3} nepers/km) obtained, we are confident that this imaginary part may be neglected to an excellent approximation. The only aspect of the leaking phenomena which conceivably could be of significance is the accumulative exponential decay represented by the factor $\exp(-k_I r)$, which is retained in subsequent calculations.

Sample input data for this second run with NCMPL < 0 are given in Fig. 11; a listing of the output values for OMEGA, VPHSE, and AMP below the original cutoff frequencies for the GR_0 and GR_1 modes of the running example is given in Fig. 12.

3.6 WAVEFORM SYNTHESIS

The final step in the waveform synthesis is to run the program INFRASONIC WAVEFORMS with input data including the information concerning the GR_0 and GR_1 modes computed as described in the preceding two sections. The essential difference between this run and the first such

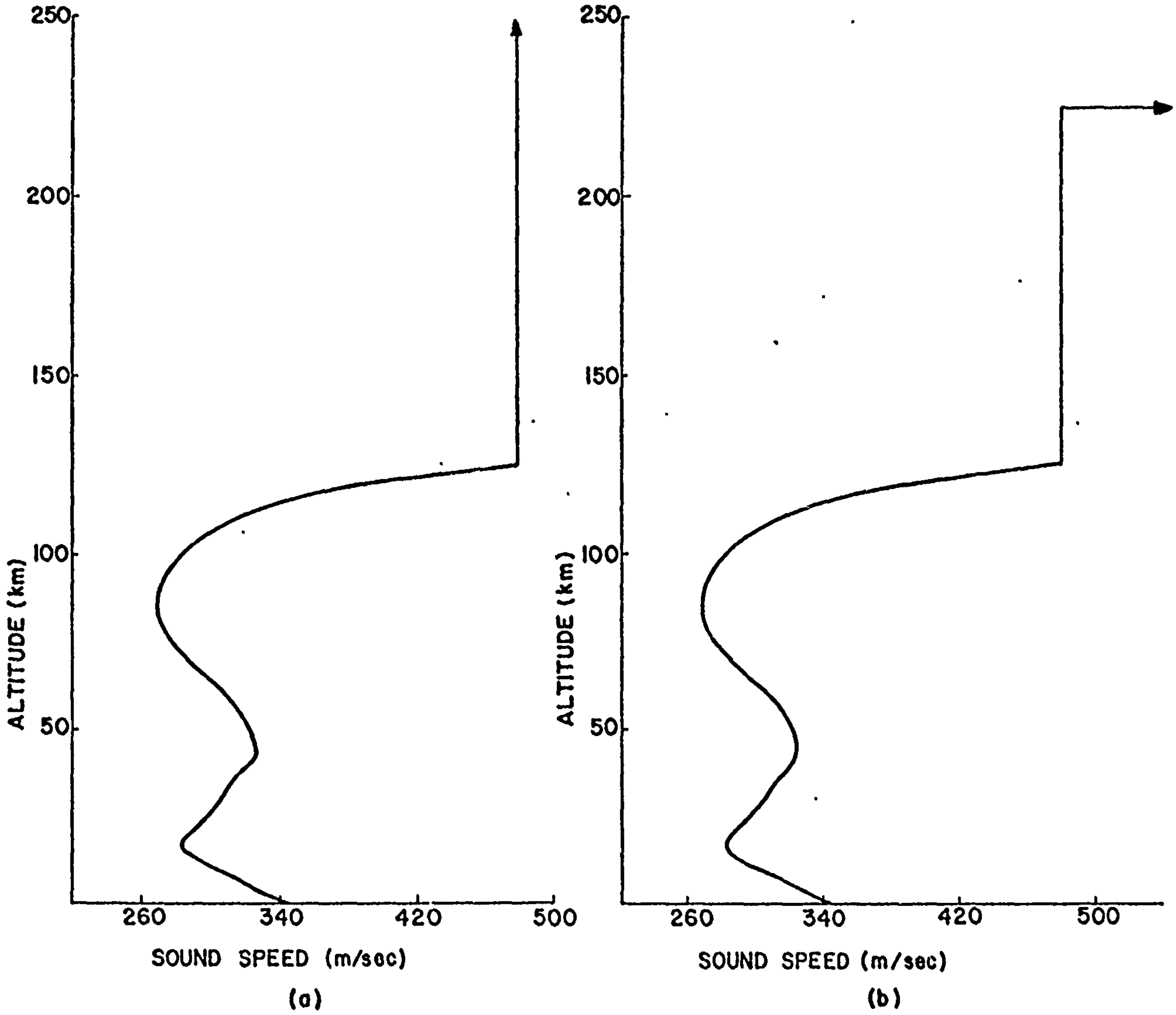


Figure 10. Two model atmosphere profiles; the first is the same as in Fig. 2; the second has the original upper halfspace replaced by a layer of finite but large thickness with a halfspace above it of extremely high temperature and sound speed. Second atmosphere is used to generate phase velocities and source free amplitudes at frequencies below nominal cutoff frequencies.

```
$NAM1 NSTART=1, NPRNT=1, NPNCH=-1, NCMPL=-1  $END
$NAM2 IMAX=25,
ZI=1.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,25.,30.,35.,40.,45.,55.,
    65.,75.,85.,95.,105.,115.,125.,225.,
T=292.,288.,270.,260.,249.,236.,225.,215.,205.,198.,205.,215.,227.,
    237.,249.,265.,260.,240.,205.,185.,184.,200.,250.,400.,570.,2.E7,
LANGLE=1,
WINDY=26*0.0,
WANGLE=26*0.0
$END
$NAM4
THETKD= 35.,
V1 = 0.18, V2 = 0.34,
ØM1 = 0.001, ØM2 = 0.02,
NØMI = 30, NVPI = 30,
MAXMØD = 8
$END
$NAM1 NSTART=6 $END
```

Figure 11. Input data to obtain phase velocity and source free amplitudes below nominal cutoff frequencies for the GR_0 and GR_1 modes.

GR_0 MODE			GR_1 MODE		
OMEGA	VPHSE	AMP	OMEGA	VPHSE	AMP
.00100	.31206	-.03102934	.00100	.28308	-.00003660
.00166	.31205	-.03101968	.00166	.28237	-.00003722
.00231	.31205	-.03100520	.00231	.28129	-.00003831
.00297	.31205	-.03098589	.00297	.27983	-.00004009
.00362	.31204	-.03096170	.00317	.27931	-.00004082
.00428	.31203	-.03093260	.00362	.27797	-.00004295
.00493	.31203	-.03089855	.00428	.27567	-.00004754
.00559	.31202	-.03085951	.00473	.27379	-.00005235
.00624	.31201	-.03081546	.00493	.27289	-.00005510
.00690	.31200	-.03076637	.00559	.26958	-.00006819
.00755	.31198	-.03071222	.00582	.26828	-.00007507
.00821	.31197	-.03065299	.00624	.26569	-.00009291
.00853	.31196	-.03062146	.00668	.26276	-.00012320
.00886	.31196	-.03058865	.00690	.26116	-.00014672
.00952	.31194	-.03051919	.00740	.25724	-.00024331
.01017	.31192	-.03044457	.00755	.25598	-.00029422
.01083	.31190	-.03036475	.00805	.25172	-.00063749
.01148	.31188	-.03027970	.00821	.25040	-.00084929
.01214	.31186	-.03018936	.00853	.24780	-.00156605
.01279	.31184	-.03009365	.00878	.24621	-.00225436
.01345	.31182	-.02999249	.00886	.24571	-.00248871
.01410	.31179	-.02988574	.00937	.24345	-.00335025
.01476	.31176	-.02977324	.00952	.24292	-.00346229
.01541	.31173	-.02965474	.01017	.24075	-.00365399
.01607	.31170	-.02952988	.01019	.24069	-.00365562
.01672	.31166	-.02939809	.01083	.23860	-.00365194
.01738	.31162	-.02925846	.01148	.23628	-.00358599
.01803	.31158	-.02910932	.01178	.23517	-.00354504
.01869	.31152	-.02894743	.01214	.23372	-.00348656
.01934	.31146	-.02876557	.01279	.23084	-.00336176
.02000	.31136	-.02854424	.01304	.22966	-.00330833
			.01345	.22758	-.00321275
			.01406	.22414	-.00305033
			.01410	.22387	-.00303760
			.01476	.21961	-.00283239
			.01490	.21862	-.00278409
			.01541	.21469	-.00259141
			.01561	.21310	-.00251310
			.01607	.20895	-.00230706
			.01621	.20759	-.00223902
			.01672	.20220	-.00196998
			.01674	.20207	-.00196321
			.01720	.19655	-.00168722
			.01738	.19420	-.00156992
			.01761	.19103	-.00141297
			.01798	.18552	-.00114281
			.01803	.18462	-.00109941
			.01831	.18000	-.00087957

Figure 12. Sample output of phase velocity and source free amplitude at frequencies below cutoff for the GR_0 and GR_1 modes corresponding to the input data of Fig. 11.

run described in Sec. 3.5 is that one sets NCMPL > 0, and that one supplies values for the parametèrs in the input list NAM51. A listing of the input data for the run, allowing for the leaking modes, and appropriate to our running example is given in Fig. 13. The phase velocities input for the GR_0 and GR_1 modes are those derived from the two computer runs described in Sec. 3.5. The source free amplitudes for these modes are supplied only for frequencies below cutoff and these are derived from the second run of Sec. 3.5. The imaginary parts of the wave number are the numbers whose computation is described in Sec. 3.5. The reason we use the phase velocities below cutoff as computed in Sec. 3.5, rather than as in Sec. 3.4, is that both calculations agree to the same order of accuracy as would be expected for the approximations inherent in the method of Sec. 3.4. Consequently, we expect the values from the computer run to be the more nearly accurate. Of course, the values of k_I have to be computed by the method of Sec. 3.4 since the computer program in its present form does not compute these directly.

In Fig.14 we show CALCOMP plots of modal and total waveforms obtained before and after the inclusion of leaking modes. (This is for our running example, 15,000 km from a 50 megaton burst at 3 km altitude, the receiver being on the ground.) One may note that the inclusion of the leaking modes eliminates the spurious precursor in the waveform and raises the amplitude of the first peak. It is also important to note that the waveform with leaking modes included begins with a pressure rise. This is what one would probably expect from intuition alone, and would also appear to be more realistic.

3.7 FURTHER EXAMPLE (HOUSATONIC)

To further explore the effects of inclusion of leaking modes, we chose the case of waveforms observed at Berkeley, California, following the Hausatonic detonation at Johnson Island on October 30, 1962. A previous comparison of theoretical and observed waveforms for this event is given in the Geophysical Journal article by Pierce and Posey.[15] This case is also the central example in the 1970 AFCRL report by Pierce and

```
$NAM1 NSTART=1,NPRNT=1, NPNCH=-1,NCMPL=1  $END
$NAM2 IMAX=24,
ZI=1.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,25.,30.,35.,40.,45.,55.,
    65.,75.,85.,95.,105.,115.,125.,
T=292.,288.,270.,260.,249.,236.,225.,215.,205.,198.,205.,215.,227.,
    237.,249.,265.,260.,240.,205.,185.,184.,200.,250.,400.,570.,
LANGLE=1,
WINDY=25*0.0,
WANGLE=25*0.0,
$END
$NAM4
THETKD=35.,
V1 = 0.15, V2 = 0.495,
ØM1 = 0.005, ØM2 = 0.1,
NØMI = 30,  NVPI = 30,
MAXMØD = 8,
$END
$NAM51  MNGR1=2, NPGR1=25, MNGRO=3, NPGRO=47,
ØMGR1=0.001,0.00231,0.00428,0.00582,0.00805,0.01017,0.01083,0.01178,
    0.01483,0.01592,0.01647,0.01706,0.01729,0.01752,0.01793,0.0181,
    0.0183,0.01864,0.01892,0.01922,0.01933,0.01935,0.01948,0.01961,
    0.01974,
VPGR1-0.28308,0.27983,0.27567,0.26828,0.25122,0.24075,0.23860,0.23517,
    0.21913,0.21034,0.205,0.19828,0.19545,0.19224,0.18621,0.1835,0.18017,
    0.17414,0.16845,0.16207,0.15954,0.15905,0.15603,0.15302,0.15,
ØMGRO=0.001,0.00231,0.00428,0.00624,0.00821,0.01017,0.01083,0.01483,0.01647,
    0.01728,0.0181,0.01892,0.01933,0.01974,0.02138,0.02177,0.02207,0.02214,
    0.02216,0.02218,0.02219,0.0222,0.02221,0.02227,0.02233,0.02253,0.02288,
    0.02302,0.0232,0.02349,0.02377,0.02404,0.02430,0.02456,0.02466,0.02483,
    0.02497,0.02511,0.02526,C.02541,0.02547,0.02575,0.02584,0.02588,0.02593,
    0.02603,0.02614,
VPGRO=0.31206,0.31205,0.31203,0.31201,0.31197,0.31192,0.31190,0.31176,
    0.31168,0.31163,0.31157,0.3115,0.31146,0.31141,0.31079,0.30991,0.30689,
    0.30539,0.30501,0.30463,0.30526,0.30417,0.30388,0.30237,0.30086,0.29483,
    0.28276,0.27772,0.27069,0.25862,0.24655,0.23448,0.22241,0.21034,0.20622,
    0.19828,0.19224,0.18621,0.18017,0.17414,0.17177,0.16207,0.15905,0.15761,
    0.15603,0.15302,0.15,
AMPGR1=-0.00003660,-0.00004009,-0.00004754,-0.00007507,-0.00063749,
    -0.00365399,-0.00365194,-0.00354504,
AMPGRO=-0.03102934,-0.03100520,-0.0309326,-0.03081546,-0.03065299,
    -0.03044457,-0.03036475,
AKIGR1=4.0E-5,9.0E-5,1.75E-4,2.4E-4,27E-4,2.5E-4,2.25E-4,1.4E-4,17*0.0,
AKICRO=3.0E-8,6.0E-8,1.2E-7,1.9E-7,2.5E-7,2.7E-7,2.3E-7,40*0.0,
$END
$NAM6  ZSCRCE=3.0,  ZØBS=0.0   $END
$NAM8  YIELD=50.E3   $END
$NAM10  RØBS = 15000.,
    TFIRST=46.2E3, TEND=52.2E3,
    DELTT=15.,
    IOPT=11
$END
$NAM1  NSTART=6,  $END
```

Figure 13. Sample input data for synthesis of infrasonic waveform including leaking modes. The data for the NAM51 input list is as derived from previous computations described in the present chapter.

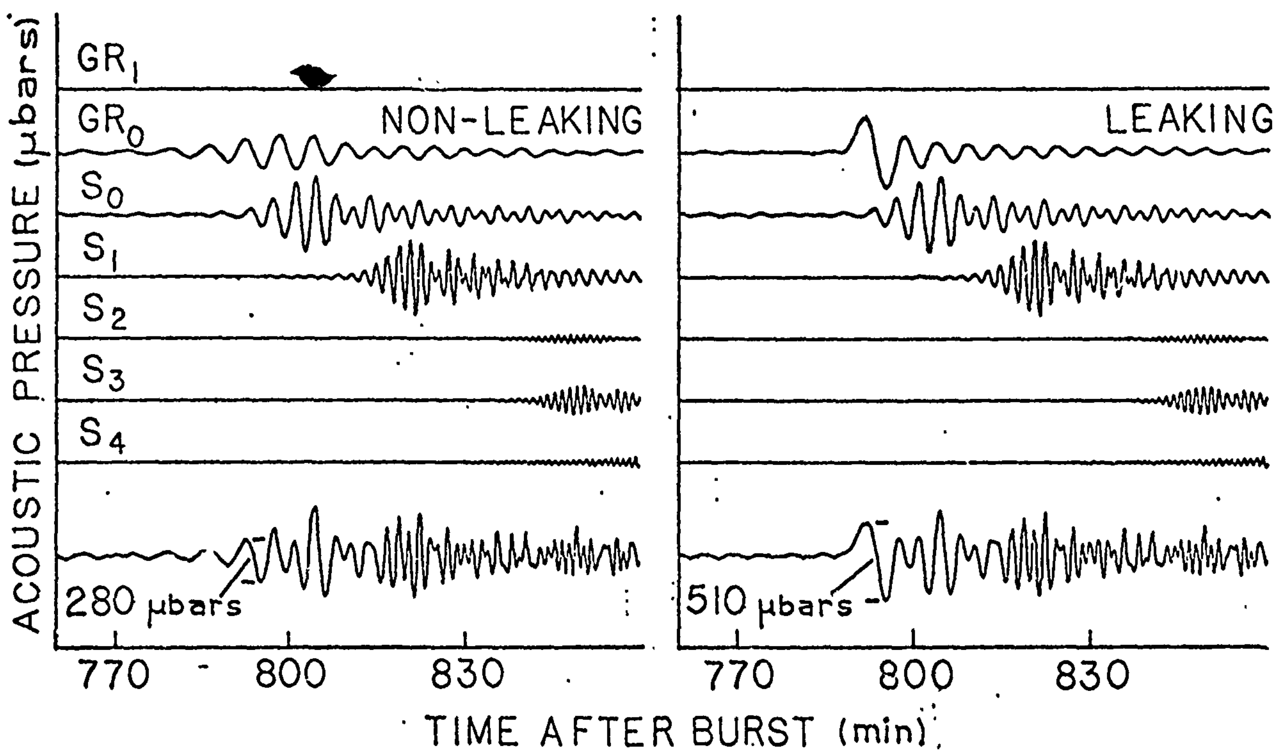


Figure 14. CALCOMP plots of modal and total waveforms before and after inclusion of leaking modes. Example is for the case of a 50 megaton burst at 3 km altitude in the atmosphere of Fig. 2; receiver is at distance of 15,000 km.

Posey[1], and is discussed within the Lamb edge mode theory context in some detail in Posey's thesis.[16]

The model atmosphere assumed for the computation is exactly the same as in Fig. 3-12 of the 1970 report, only we let the upper half space begin at 125 km (IMAX = 24). Rather than repeat the tedious calculations of the k_I for the GR_0 and GR_1 modes for this model atmosphere, we assumed that they would be essentially the same as for the running example in the previous section. Thus the steps in Secs. 3.5 and 3.6 needed only to be carried out to obtain a waveform sysnthesis.

In Fig. 15, we give comparisons of the CALCOMP plots for this event before and after the inclusion of leaking modes. One may note that the first of these does not agree with the comparable CALCOMP plots in Fig. 3-10 of the 1970 AFCRL report. This is of course because we have here taken the upper halfspace to begin at a lower altitude. This choice of where the upper halfspace begins is of little consequence when leaking modes are included, and consequently the agreement of the old computation with the leaking mode included case is quite substantial. Further, the new computation is regarded as an improvement in that the spurious initial pressure drop has been eliminated.

On the basis of the calculations described above, we have redrawn the Fig. 7 in the Geophysical Journal article which compares observed and theoretical pressure waveforms for the Housatonic-Berkeley event. This revised figure is given here as Fig. 16. The only difference is in the center waveform. The precursor is now absent and the first peak to trough amplitude has been changed from 157 μbar to 170 μbar (less than 10% increase); the remanider of the waveform is virtually unchanged. The discrepancy with the edge mode synthesis hasn't been diminished and remains a topic for future study. (It was not addressed during the present study.)

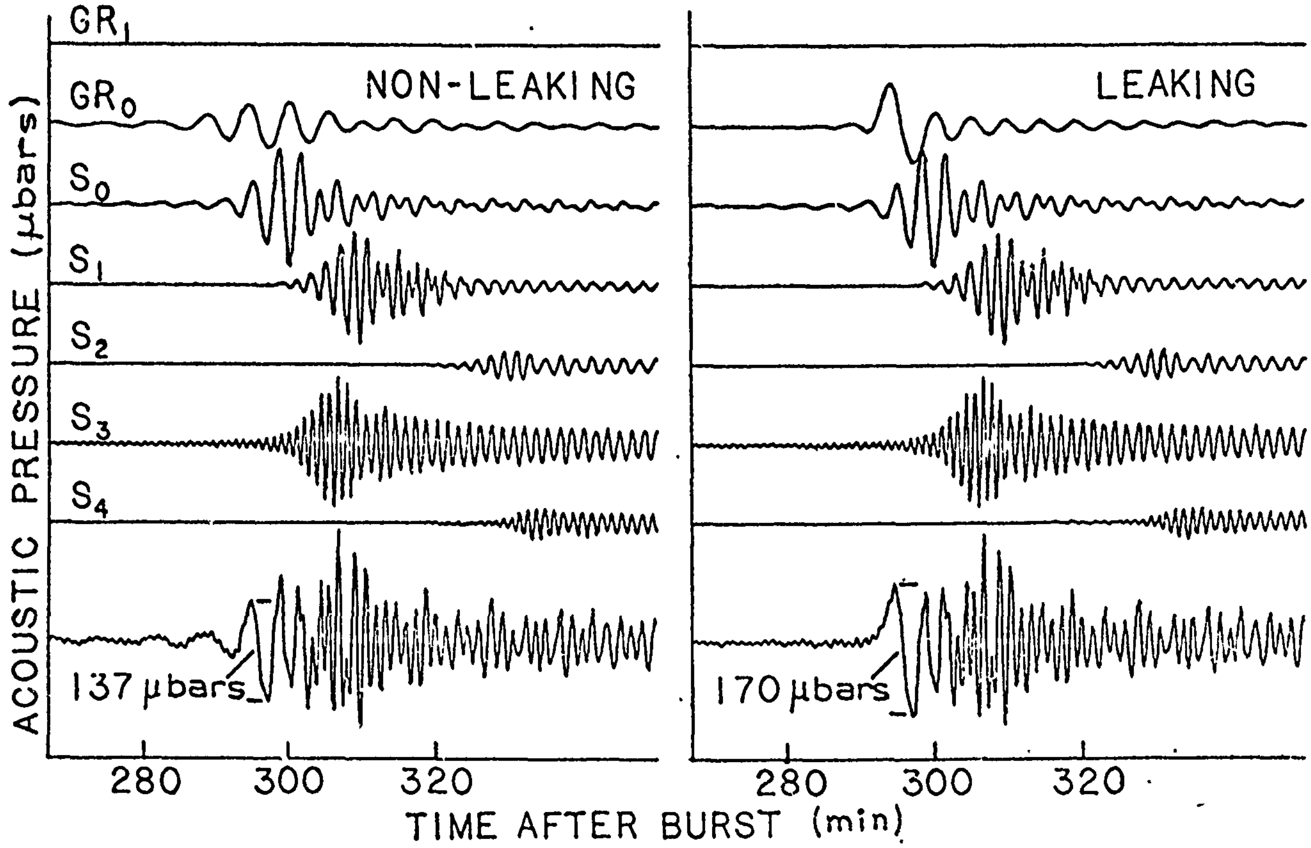


Figure 15. CALCOMP plots of modal and total waveforms before and after the inclusion of leaking modes. The ovservations were made at Berkeley, California, following the Housatonic detonation at Johnson Island on 30 October 1962. The energy yield assumed in the theoretical computations was 10 megaton. The model atmosphere is as previously used by Pierce and Posey in AFCRL-70-0134, only the upper halfspace begins at 125 km.

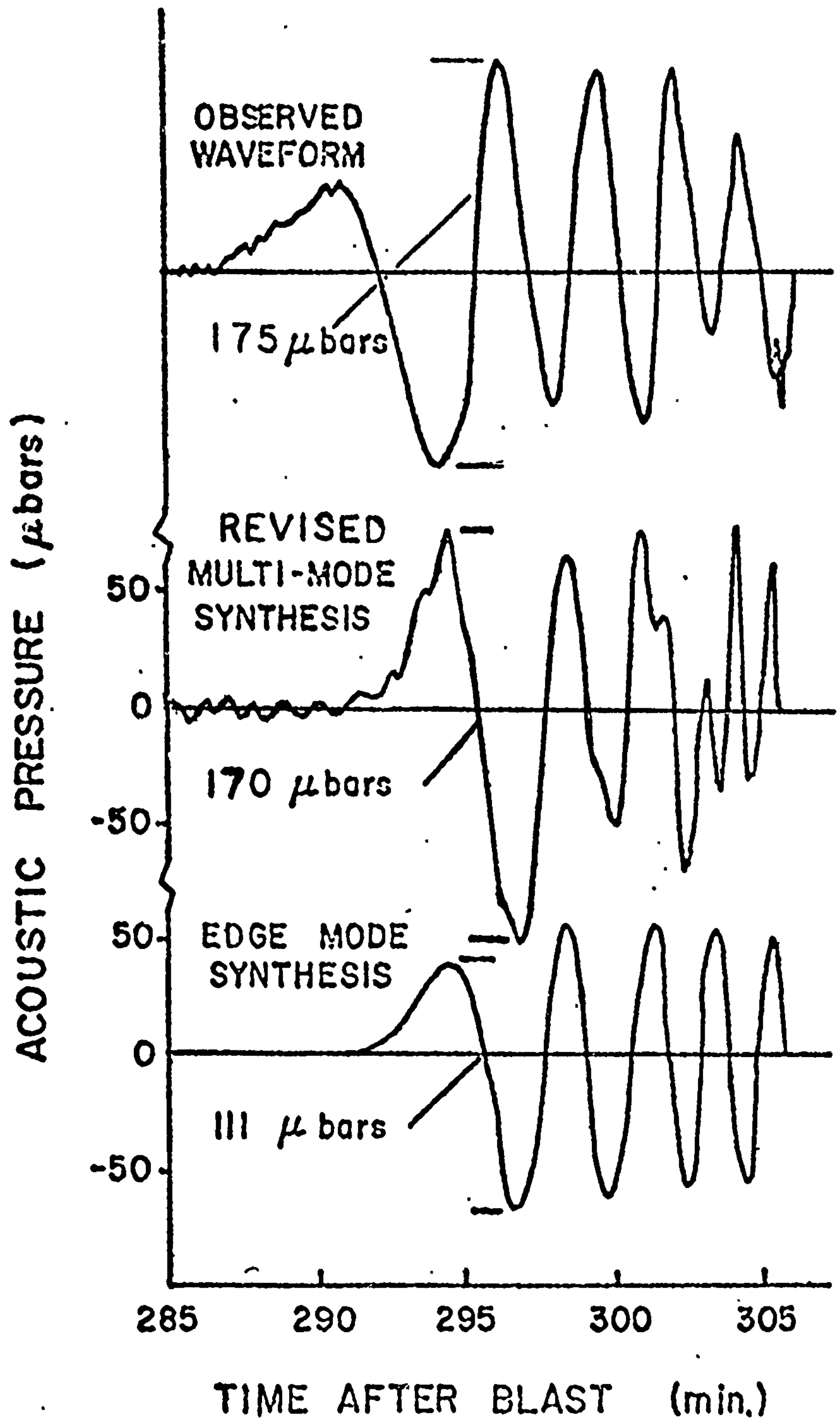


Figure 16. Observed and theoretical pressure waveforms at Berkeley, California, following the Housatonic detonation at Johnson Island on 30 October 1962. The observed waveform is taken from Donn and Shaw (1967). The energy yield assumed in the theoretical computations was 10 megatons. This is a revised version of the Fig. 7 in the 1971 paper by Pierce and Posey (Geophys. J. Roy. Astron. Soc. 26, 341-368). The original multi-mode synthesis figure has been replaced by one including leaking modes.

Chapter IV

ASYMPTOTIC HIGH-FREQUENCY BEHAVIOR OF GUIDED MODES

4.1 INTRODUCTION

Due to temperature and wind stratification, the earth's atmosphere possesses sound speed channels with associated relative sound speed minima. Fig.17 shows a standard reference atmosphere wherein two such sound speed channels are indicated; one with a minimum occurring at approximately 16 km altitude and the second with a minimum occurring at approximately 86 km altitude. Given the presence of such a channel, an acoustic ducting phenomenon can occur, as is demonstrated in Fig.18, wherein the energy associated with an acoustic disturbance can become trapped in the region of a relative sound speed minimum. It is this mechanism of ducting only that is of interest here.

In the computer program INFRASONIC WAVEFORMS, the computation of modal waveforms involves the numerical integration over angular frequency of a Fourier transform of acoustic pressure where this integration is truncated at the high-frequency end. It has been speculated that this abrupt truncation leads to the generation of what might be called "numerical noise" in the computer output. It was felt useful, therefore, to extend this integration beyond the heretofore upper angular frequency limit by means of some appropriate high-frequency approximation. In the case of an atmosphere with just one sound channel, the technique for doing this is well known and dates back to a paper published by N. Haskell[17] in 1951. Haskell's method is the W.K.B.J. (Wentzel, Kramers, Brillouin, Jeffreys) method, then in common use in quantum mechanics, although its invention dates back to Carlini[18] and Green[19] in the early 19th century.

The approximations associated with the W.K.B.J. method of solution apply to the analytical model on which the computer program is based at

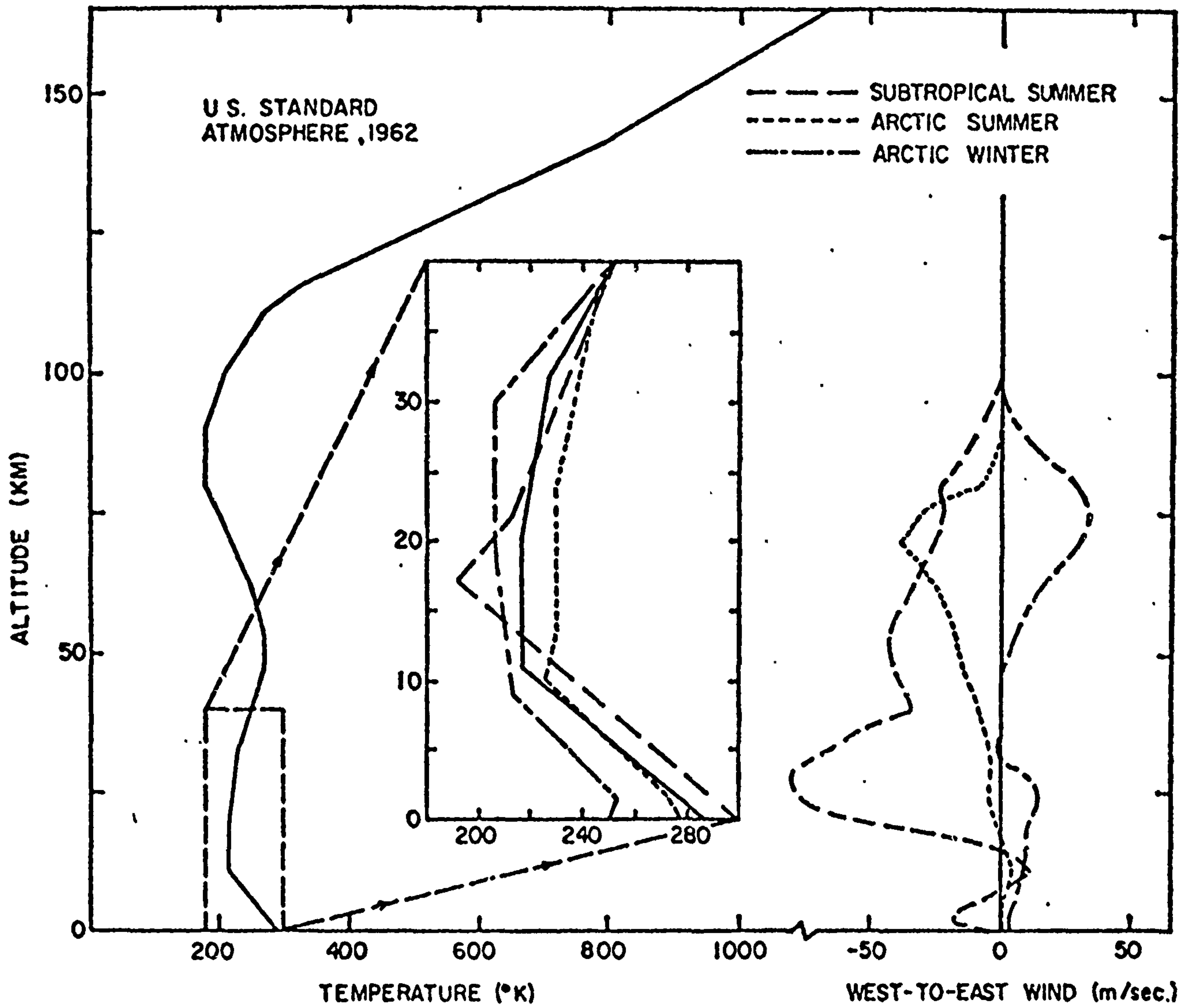


Figure 17. Temperature and wind speed versus height profiles for standard reference atmospheres. Calculations in present chapter are for U. S. Standard Atmosphere 1962 without winds. The presence of two temperature minima indicates two sound speed channels.

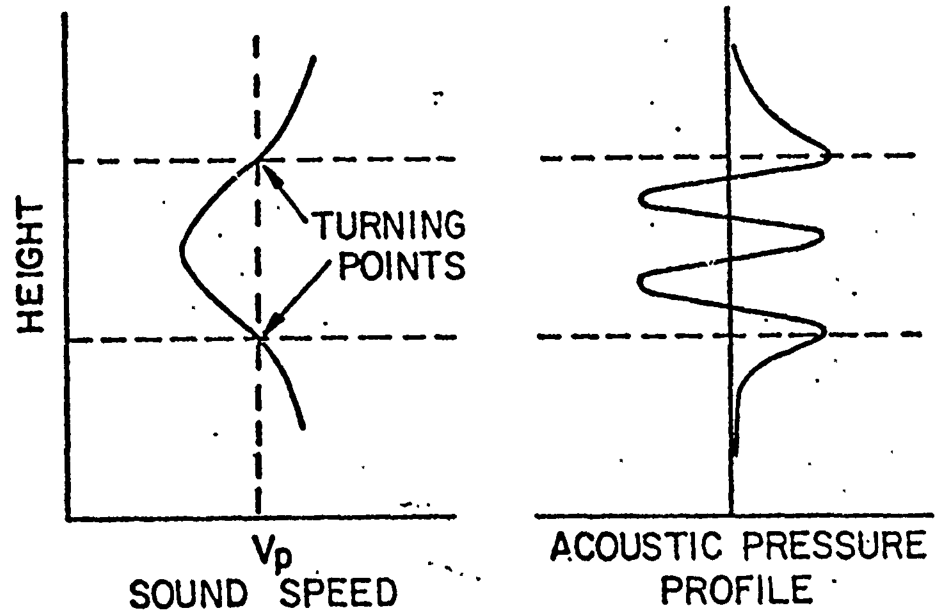


Figure 18. Sketches of sound speed versus height and acoustic pressure amplitude versus height for a guided mode illustrating the mechanism of acoustic ducting in a sound speed channel centered at a region of minimum sound speed. The energy of the disturbance may be considered as concentrated in the height region between turning points.

frequencies above approximately 0.05 radian/sec (periods less than 2 minutes). Below that limit, effects due to density stratification in the atmosphere and gravitational forces cannot be neglected. Such effects therefore are not germane to the discussion here.

The application of the W.K.B.J. method of solution to the problem of describing propagation of acoustic disturbances in an atmosphere that contains two adjacent sound speed channels has previously been discussed in the literature by Eckart,[20] who invented the simple method of seeking a W.K.B.J. model for each of the sound speed channels spearately, then combining the results rather than treating the problem with a single model. In the present chapter, Eckart's method is applied and numerically verified for the case of infrasonic waves in the atmosphere.

4.2 THE W.K.B.J. MODEL

The W.K.B.J. model for propagation of acoustic disturbances in a single sound speed channel may be considered as an approximation for the acoustic pressure divided by the square root of the ambient density, which in general may be expressed as

$$\frac{P}{\sqrt{\rho_o}} = \psi(z)e^{-i\omega t}e^{ikx} \qquad (4.1)$$

where ω is angular frequency, k is the wave number associated with the horizontal dimension x, z is altitude. Here $\psi(z)$ satisfies the reduced wave equation,

$$\left[\frac{d^2}{dz^2} + \frac{\omega^2}{c^2(z)} - k^2\right]\psi = 0 \qquad (4.2)$$

where c(z) is sound speed as a function of altitude. The W.K.B.J. approximation applies in general to all differential equations of this type if the coeffieient of ψ is sufficiently "slowly varying." It would appear in particular to be valid in the present context provided

$$\frac{c}{|\nabla c|} << \lambda \qquad (4.3)$$

where λ is some representative wavelength of interest. This approximation states that substantial changes in sound speed should not occur within distances corresponding to a typical wavelength of interest if the model is to apply.

A particular result of the W.K.B.J. approximation is that dispersion curves (v_p vs. ω) of guided modes are given by the equation

$$\int_{z_{bottom}}^{z_{top}} \left[c^{-2} - v_p^{-2}\right]^{\frac{1}{2}} dz = \frac{(2n+1)\pi}{2\omega} \tag{4.4}$$

where v_p is phase velocity, $n = 0, 1, 2, 3, \ldots$, and where z_{bottom} and z_{top} identify the lower and upper bounds of the sound speed channel, respectively. This integral is a direct result of the W.K.B.J. method of solution[21], and its numerical solution enables the plotting of dispersion curves.

4.3 COMPARISON OF DISPERSION CURVES

Particular insight into the high-frequency behavior of guided infrasonic modes was gained when the above integral was solved numerically by computer for both the upper and lower channels, the model atmosphere being that given in Fig.17. The resulting dispersion curves computed in this manner are shown in the lower portion of Fig 19. One set of curves (the dashed curves) is appropriate to the W.K.B.J. model for the lower channel and the other set (the solid curves) is appropriate to the W.K.B.J. model for the upper channel. In the upper portion of the same figure are shown again dispersion curves as generated by the computer model INFRASONIC WAVEFORMS. It should be mentioned that the computer model solves a more complex problem in the sense that the simplifications inherent in the W.K.B.J. model are not present.

As is illustrated in the lower portion of Fig.19, the two sets of dispersion curves generated by the W.K.B.J. models intersect with one another at various points. A comparison of the dispersion curves shown in both the upper and lower portions of Fig. 19 reveals that these points

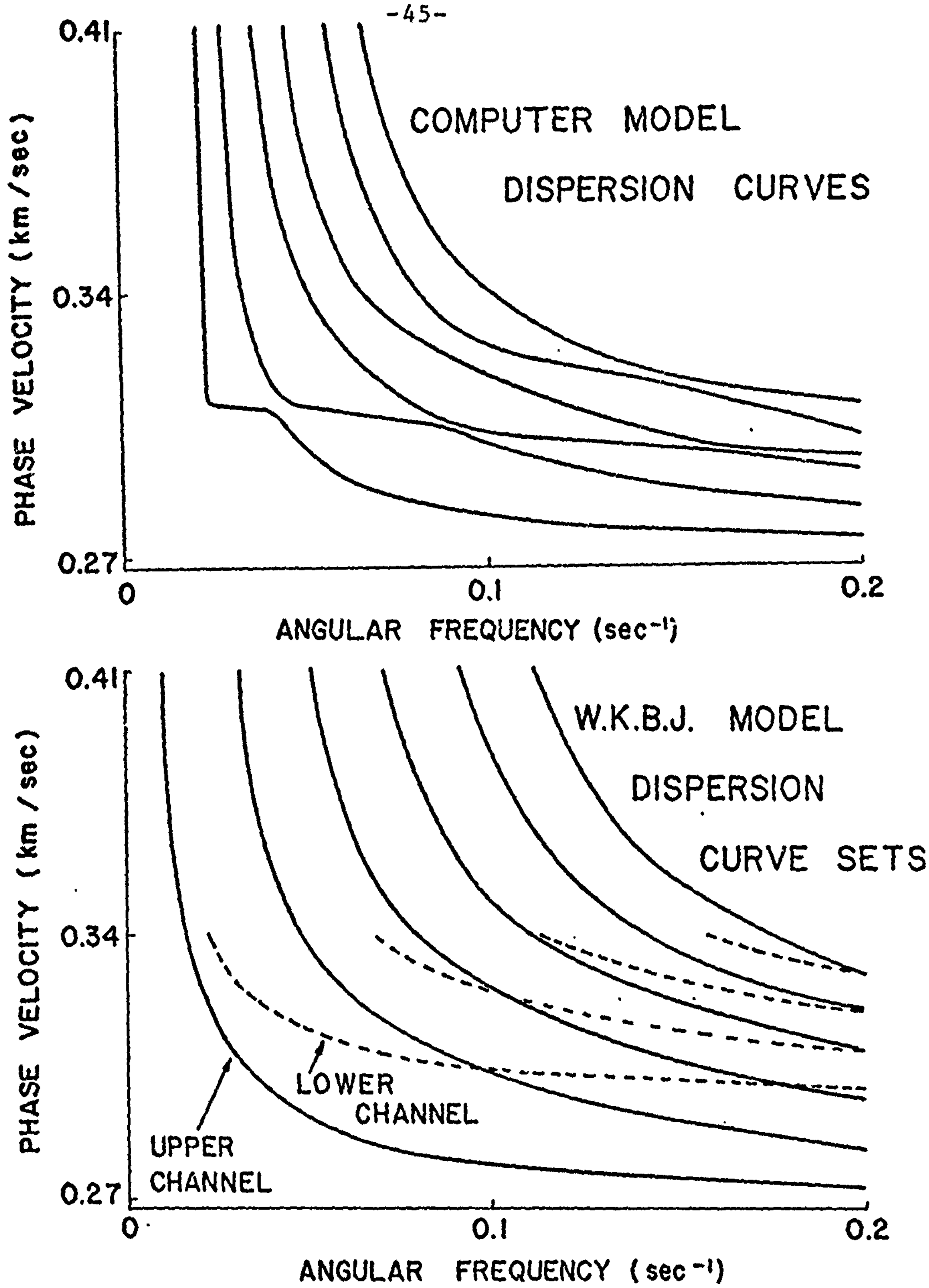


Figure 19. A comparison of theoretical guided mode dispersion curves for the U. S. Standard Atmosphere 1962. The upper set of curves were generated by full wave calculations with the multi-modal synthesis program INFRASONIC WAVEFORMS. The lower sets were obtained by applying the W.K.B.J. method to the upper sound channel (solid lines) and the lower sound channel (dashed lines), respectively.

of intersection mark regions of resonant interaction in the phase velocity-angular frequency plane between adjacent modes of the computer model. To better illustrate this observation, in the right hand portion of Fig. 20 is shown one such region of interaction with its corresponding point of intersection between two dispersion curves of the W.K.B.J. models shown to the left. It should be mentioned that the dispersion curves of the computer model never intersect with one another. An analytical explanation of this fact has previously been given by Pierce[22].

4.4 INFERENCES CONCERNING ENERGY VERSUS HEIGHT DISTRIBUTION

The above observation may be stated differently by saying that, for relatively high angular frequencies, the dispersion curve corresponding to a given mode of the computer model is comprised of portions of dispersion curves from both sets of the curves generated by the W.K.B.J. models. Two important inferences about the asymptotic high-frequency behavior of guided infrasonic modes can be drawn from this statement. First, for some frequency ranges, and depending on how dispersion curve portions match between curves of the computer model and the W.K.B.J. models, it can be inferred that the acoustic energy associated with a given mode is comprised of energy associated more with propagation of acoustic disturbances in one sound speed channel than in the other. Also, as frequency increases, this association alternates back and forth between channels. To illustrate, if, for a small range of frequencies, a portion of a dispersion curve of the computer model matches (in the phase velocity-angular frequency plane) a portion of one of the W.K.B.J. model curves for the upper channel, then that implies that, for that mode and for that small frequency range, the acoustic energy density associated with that mode is greater in the upper channel than in the lower channel. Secondly, in the standard reference atmosphere, the sound speed minimum for the upper channel is less in magnitude than the sound speed minimum for the lower channel. It can be inferred, therefore, that those acoustic disturbances for which phase velocities are less in magnitude than the sound speed minimum for the lower channel are associated more with acoustic energy trapped in the upper channel than in the lower channel, and thus, for this reason, do not contribute significantly to the acoustic energy at the ground. This inference implies that care must

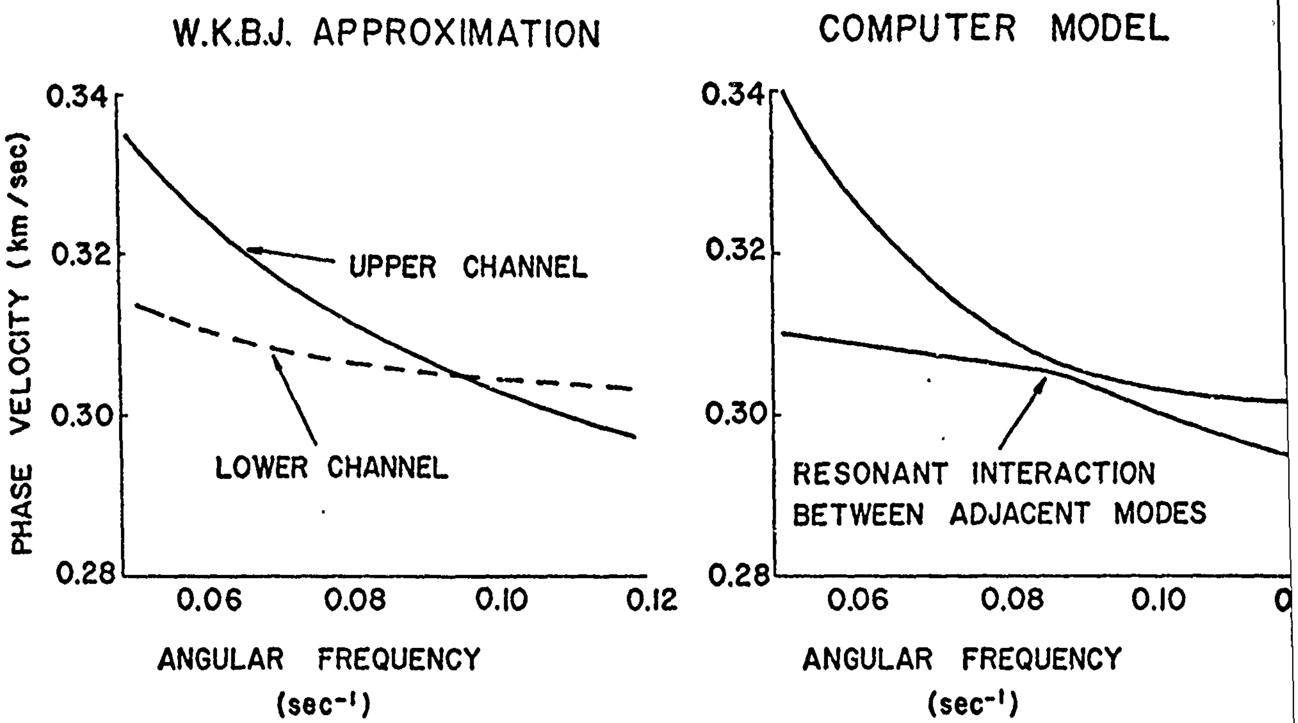


Figure 20. A detailed (blown-up) plot of a section of Fig. 19 showing a region of resonant interaction between two modes, one ducted in the upper channel, the other ducted in the lower channel. The full wave calculation (computer model) indicates that the two modes interact such that the actual dispersion curves do not cross, but indicates that the W.K.B.J. and computer model curves are nearly the same except in the region of resonant interaction.

be taken as to which modes are chosen to superpose in the attainment of the final pressure waveform at the ground, as some may not contribute.

4.5 IMPLICATIONS FOR WAVEFORM SYNTHESIS

In the previous synthesis of guided pressure waveforms at long distances, the acoustic modes were numbered in order of increasing phase velocity (i.e., S_0, S_1, S_2,..., etc.) and the sum over modes was truncated at a finite maximum number of modes. The analysis given here indicates that this may be a very poor approximation for synthesizing high frequency portions of waveforms observed near the ground since there is always some frequency above which the first, say, N modes all correspond to channelling in the upper sound speed channel.

The preferable alternative would appear to be (for synthesis of ground level arrivals from sources below 50 km altitude) to ignore the upper sound speed channel completely for frequencies above, say, at least 0.2 rad/sec (possibly 0.1 rad/sec) corresponding to periods below at most 30 sec (possibly 1 min). The dispersion curves could then be taken as given by the W.K.B.J. approximation and the mode amplitude versus height profiles could be computed by the method outlined by Haskell. The dispersion curves and amplitudes so computed would fit directly into the general scheme outlined by Pierce and Posey[1] which forms the theoretical basis for the current version of INFRASONIC WAVEFORMS.

Chapter V

EXTENSION OF INFRASONIC WAVEFORMS TO INCLUDE DISTANCES BEYOND THE ANTIPODE

5.1 INTRODUCTION

Previous theoretical considerations incorporated into the digital computer program INFRASONIC WAVEFORMS restricted synthesis to waves that had traveled less than one-half the distance around the earth. The purpose of this chapter is to further exemplify techniques to enable computer synthesis of acoustic-gravity pressure waveforms at points whose distances are greater than halfway around the world from a nuclear explosion. Extension of prior theory shows that for wave propagation past a point on a spherical earth, one-half the great circle distance away from the point of detonation (i.e., the antipode), a phase shift of $\pi/2$ radians to the Fourier transforms of each modal wave is incurred. Modification to the computer program necessitates the reinterpretation of the great circle distance r, the inclusion of the $\pi/2$ phase shift, and a modification to the earth curvature correction factor. Computations are presented for pre and post antipodal waveforms.

5.2 THEORETICAL CONSIDERATIONS FOR POST ANTIPODAL WAVEFORMS

In considering acoustic-gravity waves that have passed beyond the antipode, certain specific definitions for the various waveforms must be adopted. To an observer located on the surface of a spherical earth between the source and the antipode the pressure waveform that is first observed is the direct arrival or A_1 arrival. The A_1 arrival has traveled the shortest great circle distance r to reach the observation point. The next waveform observed at the above observation point is the A_2 or antipodal arrival. The A_2 arrival has traveled the longer great circle distance from the explosion point around the globe passing through the antipode to reach the observation point. The A_3 arrival is the A_1 pressure waveform that has traveled completely around the globe with respect

to the observation point. Further arrivals exist but are not considered here. The distance r is measured in kilometers and is the great circle distance measured from the detonation point to the final observation point. Figure 21 shows some typical pressure waveforms recorded in suburban New York for the Russian explosion of 58 megatons at Novaya Zemlya on 30 October 1961.[23]

Previous numerical syntheses of acoustic-gravity waveforms have only considered direct arrivals. The extension of this theory to include waveform prediction for antipodal arrivals is described here. An investigation of a small region of the earth's surface in the vicinity of the antipode where prior theory breaks down yields certain waveform characteristics that enable waveform synthesis to be extended to ranges past the antipode. By taking the antipodal region smaller in area than say 1/100th of the earth's area as a whole we can consider this region to be flat. Then the equation governing propagation of any frequency in any guided mode near the antipode is the cylindrical wave equation in the form of

$$\partial^2 F/\partial r_A^2 + (1/r_A)\partial F/\partial r_A - (1/V_p^2)\partial^2 F/\partial t^2 = 0 \tag{5.1}$$

where F would represent the r_A and t dependent part of the integration kernal for synthesization (i.e., integration over frequency of any given modal waveform where the height dependent part is omitted here). The quantity V_p is the corresponding phase velocity. The assumed circular symmetry of the wave about the antipode is inherent in the absence of the angular derivative terms in the above equation. The distance r_A is measured positive out from the antipode. The wave solution to Eq. (5.1) for the total acoustic pressure p and small r_A can be written for time t as

$$F \cong DJ_o(kr_A)\cos(\omega t+\varepsilon) \tag{5.2}$$

For the above, $k = \omega/V_p$ represents the horizontal wave number, ω the angular frequency, and ε some phase angle. The quantity D is some arbitrary constant while $J_o(kr_A)$ is the Bessel function of zero order.

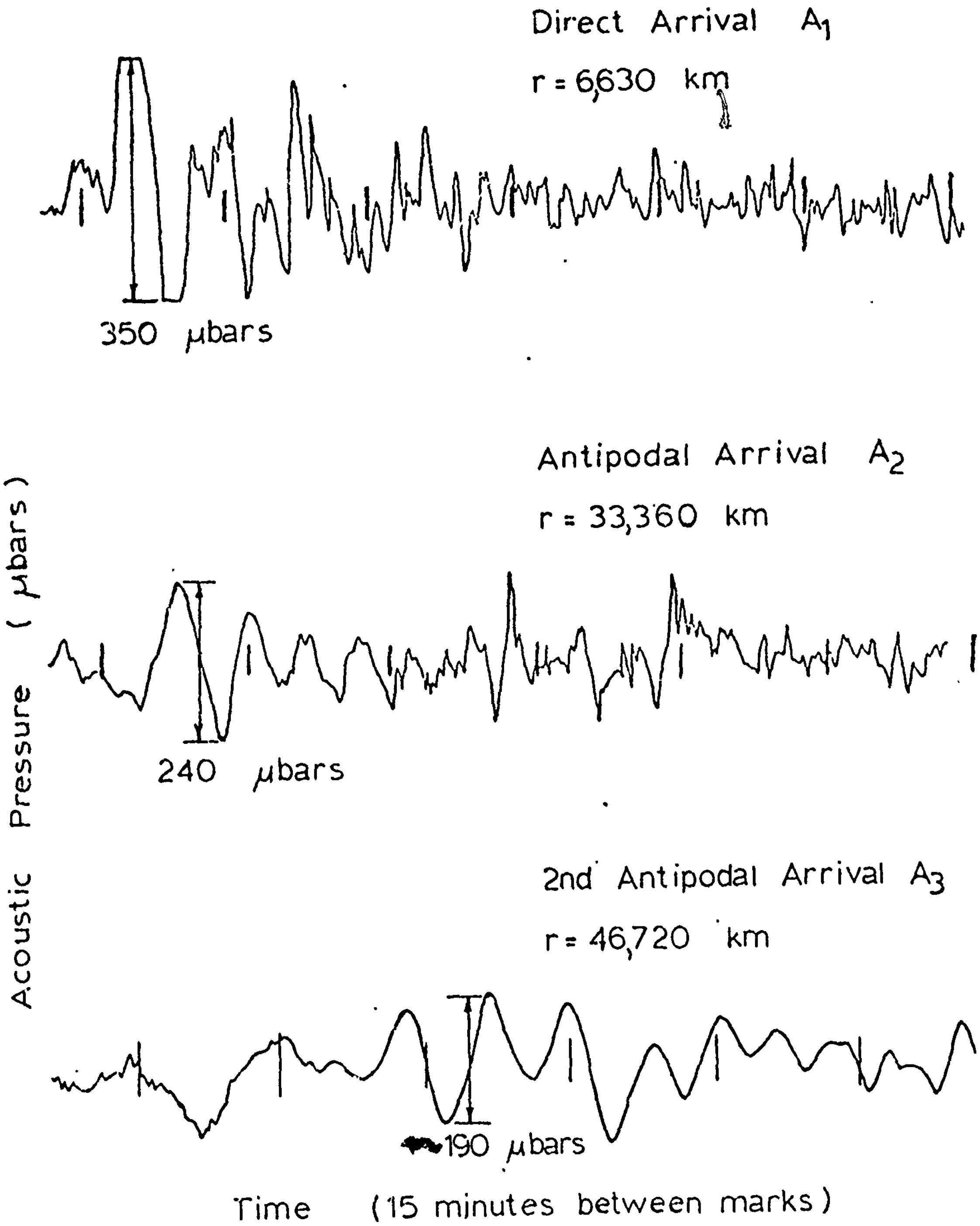


Figure 21. Infrasonic pressure waveforms recorded in suburban New York following the detonation of a 58 megaton yield nuclear device in Novaya Zemlya USSR on 30 October 1961. [Extracted from Donn and Shaw, Rev. of Geophys. 5, 53-82 (1967).]

When r_A is suffienciently large (i.e., greater than three wavelengths) a solution for the total acoustic pressure p can be considered as a sum of ingoing and outgoing waves with respect to the antipodal region. The asymptotic solution for large kr_A can be written for time t as

$$F = A(r_A)^{-1/2}\cos(\omega t+kr_A+\phi_{in}) + B(r_A)^{-1/2}\cos(\omega t-kr_A+\phi_{out}) \tag{5.3}$$

In Eq. (5.3) ϕ is some phase angle while ω and k are as previously defined. The plus sign in the argument of the cosine denotes an ingoing wave. Equation (5.3) is not defined at $r_A = 0$ and, as r_A approaches zero, wave amplification is predicted. Figure 22 illustrates waveform amplification approaching the antipode for three different values of r for a ten megaton nuclear explosion. The antipode is reached when r = 20,000 km.

Realizing that Eqs. (5.2) and (5.3) should represent the same presure waveform at large r_A we can now show the existence of a phase difference between waveforms approaching and leaving the antipode. For large r_A, the Bessel function $J_o(kr_A)$ can be represented by its asymptotic approximation such that Eq. (5.2) becomes

$$F = D(2/\pi r_A k)^{1/2}\cos(kr_A-\pi/4)\cos(\omega t+\varepsilon) \tag{5.4}$$

or with the aid of trigonometric identities as

$$F = \frac{1}{2}D(2/\pi r_A k)^{1/2}\ [\cos(\omega t+\varepsilon+kr_A-\pi/4) + \cos\ (\omega t+\varepsilon-kr_A+\pi/4)] \tag{5.5}$$

Equating (5.3) to (5.5) then requires that

$$A = B = D/(2\pi k)^{1/2} \tag{5.6a}$$

$$\phi_{in} = \varepsilon - \pi/4 \tag{5.6b}$$

$$\phi_{out} = \varepsilon + \pi/4 \tag{5.6c}$$

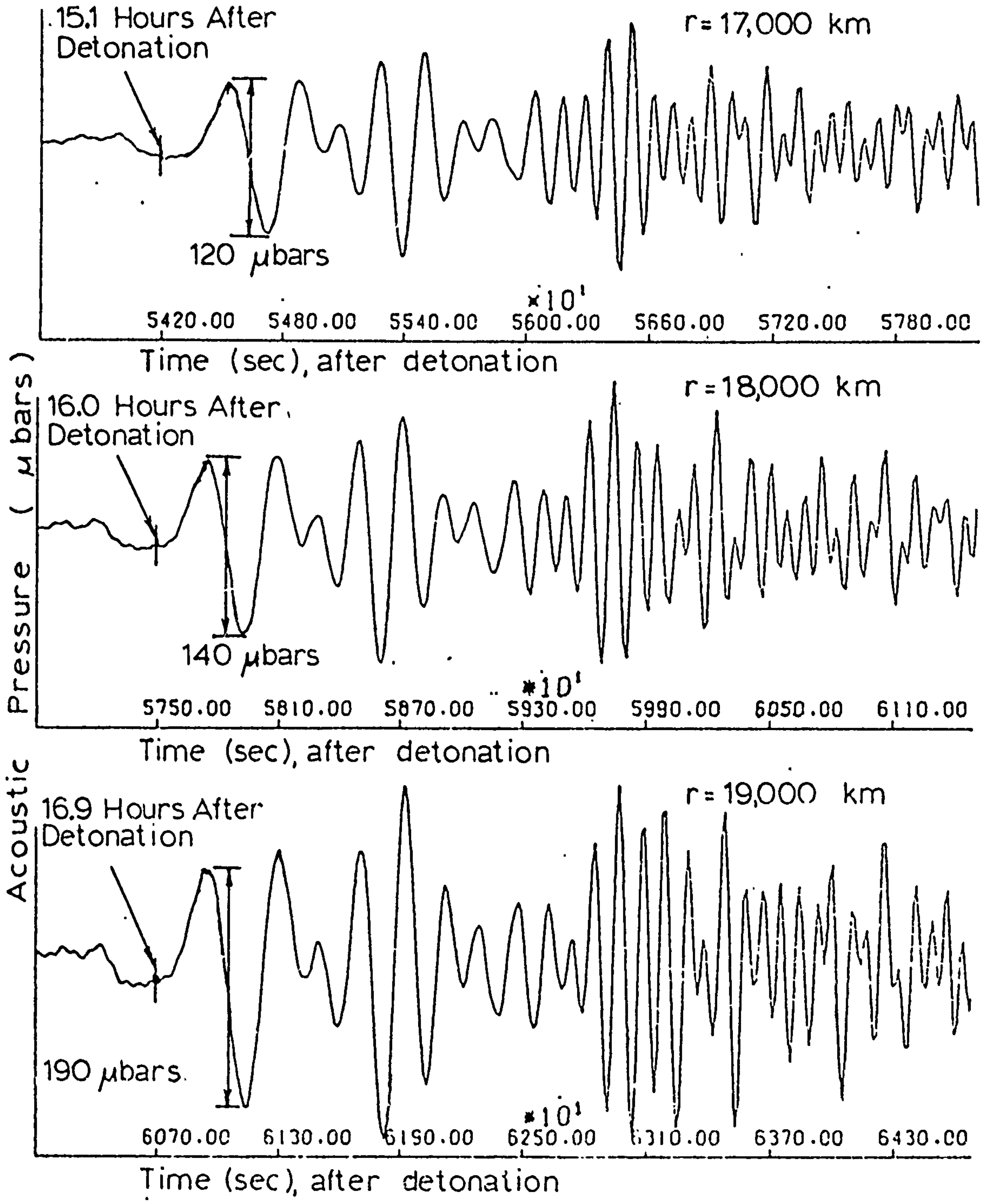


Figure 22. Theoretical pressure waveforms of a pulse propagating towards the antipode (corresponding to a great circle distance r of 20,000 km). Computations presented are for a 10 megaton burst in a standard atmosphere without winds. Note the amplification in amplitude for values of r successively closer to 20,000 km.

so

$$\phi_{out} = \phi_{in} + \pi/2 \quad (5.7)$$

The latter shows that a pressure waveform undergoes a phase shift of 90 degrees. Based on this knowledge the computer program has been altered to synthesize pressure waveforms for the A_2 arrival that passes through the antipode.

5.3 MODIFICATIONS TO INFRASONIC WAVEFORMS FOR POST ANTIPODAL WAVEFORMS

Waveform synthesis for ranges beyond the antipode necessitates only minor adjustments to the computer program. By considering the theoretical development of Brune, Nafe, and Alsop (1961)[24] for circular spreading of waves over a spherical surface of radius r_e (i.e., r_e = 6374 km for earth) the amplitude correction factor for the curvature of a spherical earth, appearing in subroutine TMPT, is altered for post antipodal waveforms by replacing the term $\sin(r/r_e)$ by its absolute magnitude, where r is interpreted as the total distance the wave has traveled from the point of detonation. For post antipodal arrivals considered here r would be between πr_e and $2\pi r_e$ kilometers. The earth curvature correction factor in subroutine TMPT appearing as

```
CF = (1./(6374. * SIN(RAD)))**0.5
```
(5.8)

is replaced for post antipodal waveforms by

```
CF = (1./(6374.*ABS(SIN(RAD))))**0.5
```
(5.9)

where ROBS = r and

```
RAD = ROBS/6374.
```
(5.10)

To accomodate the change in phase as the waveforms pass through the antipode two computer cards of the form

```
PH2 = PH2 + 1.570796
```
(5.11)

are inserted in the deck listing of subroutine TMPT after lines 160 and 177.

After incorporating the above modifications into subroutine TMPT the computer program was then utilized to synthesize various theoretical waveforms. Using the Soviet shot of 30 October 1961 as the source, a phase shift upon passing through the antipode is exhibited in Fig. 23 for two observation ranges of a synthesized pressure waveform. Further dispersion beyond the antipode of the pressure waveform is shown in Fig. 24 for a ten megaton explosion. A comparision of antipodal arrivals for a computer synthesized pressure waveform and a microbarograph recorded by Donn and Shaw in suburban New York[5] for the 58 megaton Soviet test is presented in Fig. 25. Considering the scattering in waveforms that can occur at such large arrival distances, it is not unreasonable to say that the amplitudes and typical periods of the two plots are of the same order of magnitude.

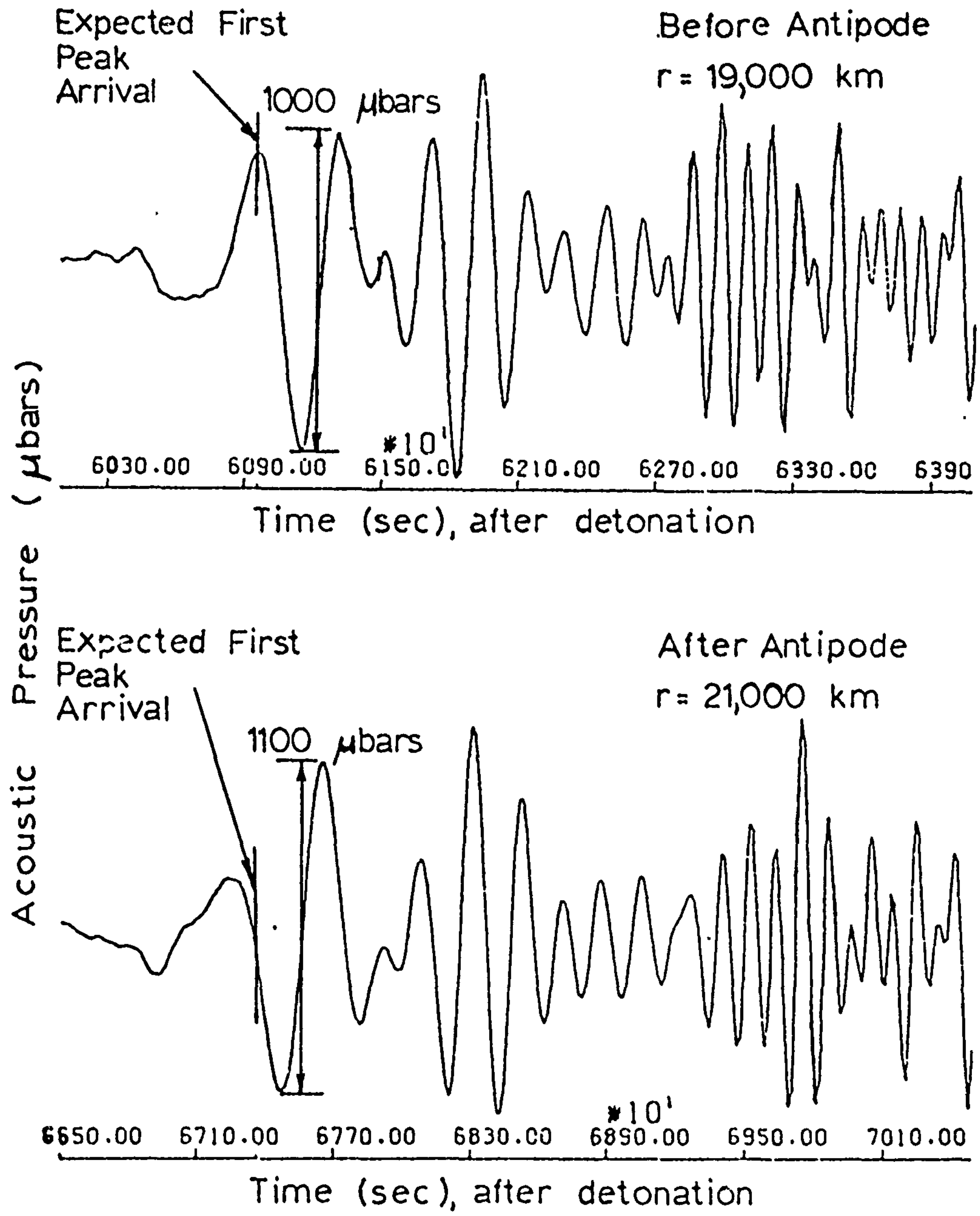


Figure 23. Theoretical pressure waveforms just before (great circle distance r of 19,000 km) and just after (r of 21,000 km) passing through the antipode (20,000 km). The $\pi/2$ phase shift after the antipodal passage is evidenced by the second figure. Time of expected first peak arrival derived from linear extrapolation of computed time of first peak arrival versus great circle distance for $r<20,000$ km to case of $r>20,000$ km. Source is the 58 megaton Soviet test in Novaya Zemlya.

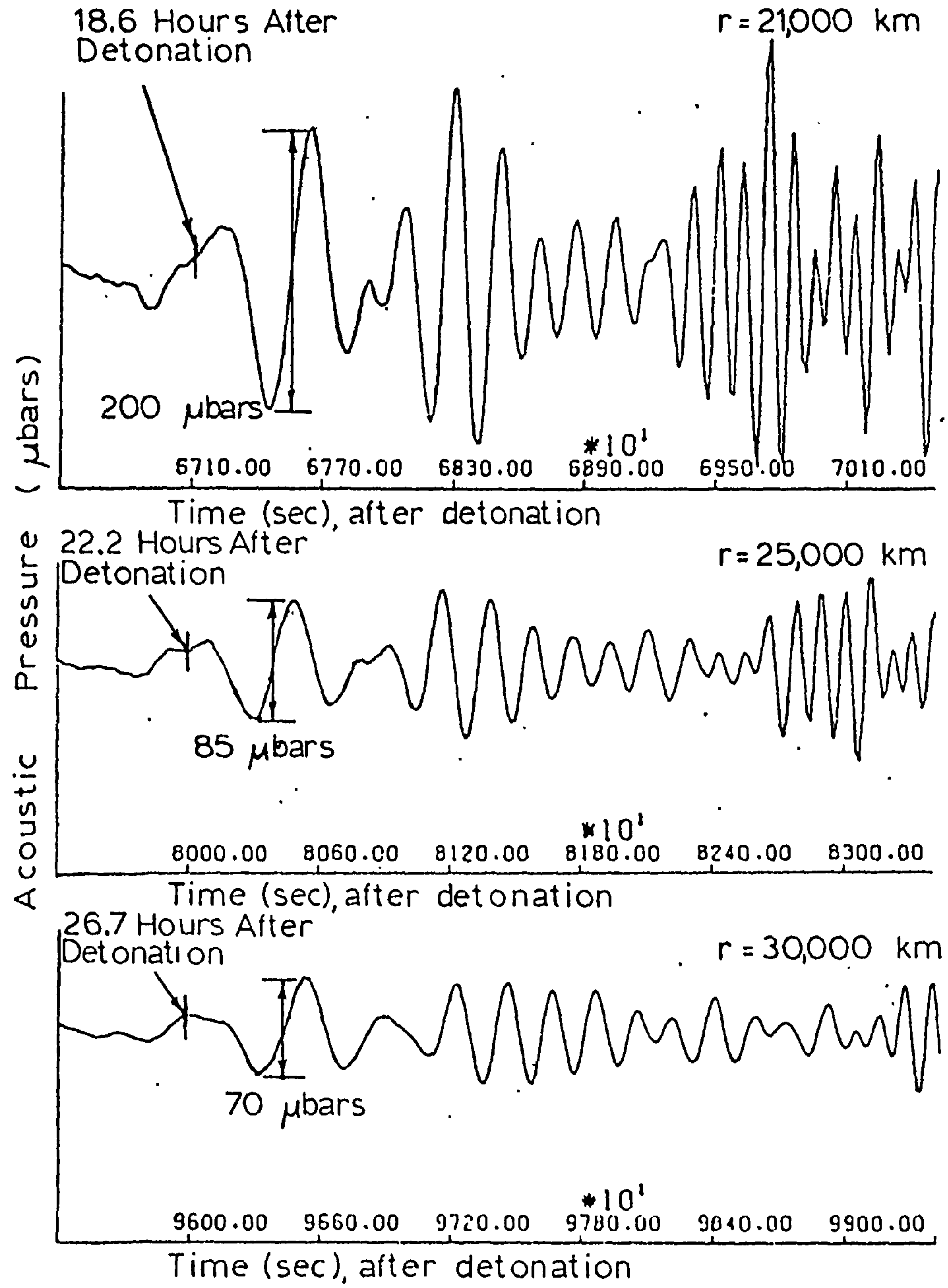


Figure 24. Theoretical pressure waveform for a pulse propagating away from the antipode. Decrease of amplitude and increased frequency dispersion occurs with increasing great circle distance r. The source is a 10 megaton nuclear explosion in a standard atmosphere without winds.

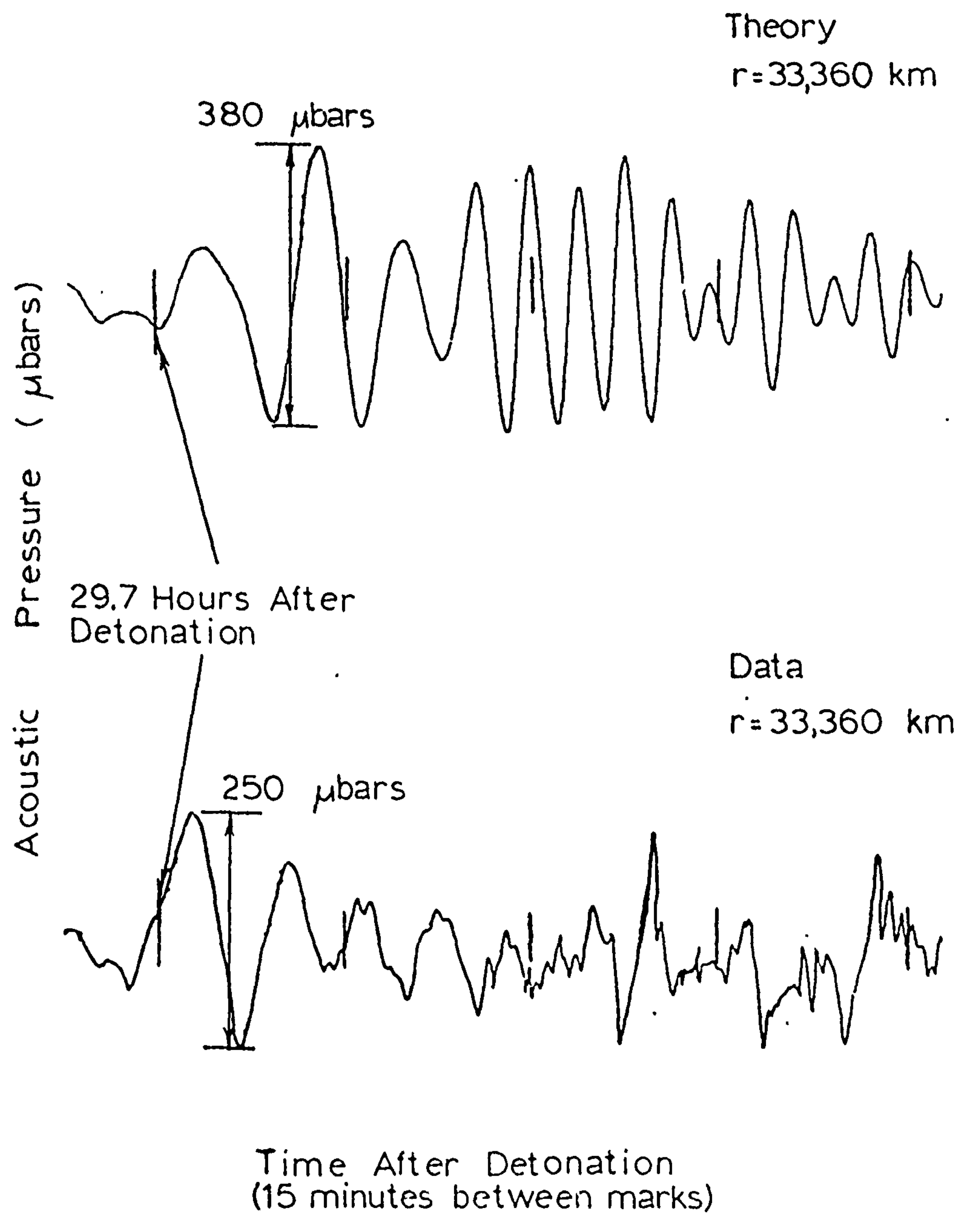


Figure 25. A comparison of theoretical and observed antipodal (A_2) arrivals for pressure wave recorded in suburban New York following the detonation of a 58 megaton yield nuclear device in Novaya Zemlya USSR on 30 October 1961. Note that the amplitude scales for the two records are not the same. Observed waveform taken from Donn and Shaw, Revs. of Geophys. 5, 53-82 (1967).

Chapter VI

CONCLUSIONS AND RECOMMENDATIONS

6.1 REMARKS CONCERNING INFRASONIC WAVEFORMS

The new version of INFRASONIC WAVEFORMS contained in this report (Appendix A) allows for the computation of waveforms which have propagated past the antipode and for the computation of waveforms including leaking modes. Our remarks here concentrate on the latter modification.

If one chooses a model atmosphere in which the sound speed is constant above some arbitrary large height, it is inevitable that the GR_0 and GR_1 modes should have lower cutoff frequencies and be leaking below that altitude. Beyond a certain point, one would expect that the computations should be independent of this choice of height, provided the analysis were carried through with some degree of exactitude. If there were a genuine sensitivity, this would indicate that these modes carry an appreciable fraction of their energies at high altitudes and this would in turn suggest that the neglect of physical dissipative mechanisms (such as viscosity and thermal conduction, Joule heating, etc.), which increase dramatically at extremely large heights for the frequencies of interest here, is not a valid approximation.

The reason we cannot take the bottom of our upper halfspace to be arbitrarily large is that some modal height-amplitudes decrease exponentially at large altitudes. This exponential decrease implies that, if one attempts to calculate the transmission matrix [R] connecting variables at the bottom of the upper halfspace to those at the ground, then the elements of [R] are going to be extremely large and the mathematical theorem that the determinant of [R] be 1, while true in principle, is not going to be satisfied for the actual numerical values computed because of the loss of significant figures. The net result is such large fluctuations in the eigenmode dispersion function due to round-off errors that it is impossible to determine its roots. This problem

always arises at sufficiently high frequencies when the upper halfspace bottom is taken too high.

In Chapter III, a simple expedient for circumventing this difficulty is implicitly described. One uses one atmosphere for low frequencies, another atmosphere for higher frequencies. The atmosphere for the higher frequency calculations has its halfspace beginning at, say, 125 km altitude while the atmosphere for the lower frequency calculations has its upper halfspace beginning at, say, 225 km. Given the premise that, for the GR_0 and GR_1 modes (which appear to be the only modes for which we have problems at low frequencies), the energy is ducted below 125 km, the temperature above 225 km can be made as large as one desires without changing the answers. Thus one simply chooses this temperature to be so large that the lower cutoff frequencies for the two modes are, for all practical purposes, zero. In this manner one can construct the phase velocities and source free amplitude functions versus frequency for these modes down to arbitrarily small frequencies.

Another question is whether or not the k_I (imaginary part of wavenumber) for the leaking modes are physically meaningful. They obviously would be meaningful were the actual atmosphere terminated by an upper halfspace and were there no physical dissipation mechanisms. However, the actual atmosphere is more complicated than this model and one has to accept the fact that (1) an approximate atmosphere is going to give rise to approximate answers and (2) that the values of the k_I are going to depend on the choice of the bottom height of the upper halfspace. Thus the k_I are really somewhat arbitrary. Fortunately, the values of the k_I so derived are very small, at least for the example we have numerically carried out, that the computed waveforms are almost the same as if the k_I were identically zero.

With the above remarks in mind, it is recommended that the calculations of the k_I for the GR_0 and GR_1 modes below cutoff <u>not be carried out</u> in the synthesizing of waveforms. Rather, one should either set the k_I for frequencies below cutoff as given in our numerical example or to 2×10^{-10} (i.e., for all intents and purposes, zero). The reason the k_I

should not be set identically to zero is that the computer program uses the nonzeroness of k_I as a flag to decide whether to look for an input value of AMP (source free amplitude) or to compute the number internally (it can't do this at frequencies below cutoff and will consequently return AMP = 0). While this may seem a rather simple thing to do, considering the elaborate mathematical theory developed[2] in Scientific Report No. 1, the analysis and computations which preceded the formulations of this recommendation were necessary, if only to establish that the procedure has some rigorous mathematical basis.

In any event, it is evident that one must and should include contributions from the frequencies below the nominal low frequency cutoff (determined by the upper halfspace) if one is to adequately synthesize the initial portions of waveforms. The present report shows how this may be done. The procedure, although requiring several (three, in general) runs of the program rather than just one run to accomplish this, is relatively straightforward. It is obviously feasible to automate this so that only one run is necessary, but the time limitations of the present study precluded our doing so.

6.2 DISCREPANCY WITH LAMB EDGE MODE THEORY

It was hoped that the inclusion of leaking modes into the multimode synthesis would eliminate the discrepancy between the numerical predictions of the Lamb edge mode theory and the multi-mode theory. It is evident, however, from Fig. 16 in the present report that this has not turned out to be the case. The cause of the discrepancy has not been resolved and time limitations precluded its resolution. There is always the possibility that either program may have a mistake. However, barring this, it should be pointed out that the modified multimode theory should be the more nearly correct. The Lamb edge mode theory[15] contains a number of approximations which the multi-mode theory does not contain. Consequently, it is recommended that the multi-mode model as modified here be used in preference to the Lamb edge mode model.

The relative simplicity of the edge mode model still retains an intrinsic appeal and, consequently, it is recommended that some future effort be expended in revising the model (possibly by including higher order terms in the dispersion relation) such that the discrepancy is resolved.

6.3 GUIDED MODES AT HIGHER FREQUENCIES

The procedure outlined in Chapter IV for using a modified W.K.B.J. approximation to order the modes and to compute modal parameter at high frequencies looks eminently feasible and is recommended for inclusion into the multi-mode synthesis program INFRASONIC WAVEFORMS. Although, again, time limitations precluded this, we regret not having done so in the present study. The motivation for doing this, however, is not as strong as for the low frequency modifications because the commonly available data in the open literature is markedly poor as regards high frequency arrivals. If and when such a modification is carried out, one should ideally have appropriate data with which to compare the numerical predictions.

Another problem is that there is some question as to whether a multmodal theory with a finite number of modes (even when judiciously selected) can ever adequately synthesize higher frequency arrivals. In many respects, we believe that an appropriate modification of a geometrical acoustics theory would be preferable.

6.4 GEOMETRICAL ACOUSTICS MODEL

The geometrical acoustics model described[3] in Scientific Report No. 2, although still incompletely developed, appears to hold considerable promise for the understanding of higher frequency arrivals. We know now how to take the edge mode into account and how to handle the problem of caustics. Problems of aretes, lacunae, and wave diffusion from channel to channel still remain, but we believe these can be overcome with only a modest amount of additional theoretical effort.

The ultimate objective of the analysis should be to develop the simplest possible theory sufficient to explain and interpret available data. In this respect, we would suggest that both the multi-mode and geometrical acoustical models, while perhaps more elaborate than should be ideally required, could be used as research tools to conduct numerical experiments which test simpler models. The statistical models developed by P. Smith[25] for underwater acoustics appear especially attractive in this regard and we believe that one should be able to test his models using the geometrical acoustics model described in Scientific Report No. 2. Also, the types of numerical experiments envisioned should provide the inspiration and support required to refine Smith's models such that they be capable of a more nearly precise description of infrasonic waveforms.

REFERENCES

1. A. D. Pierce and J. W. Posey, "Theoretical Prediction of Acoustic-Gravity Waveforms generated by Large Explosions in the Atmosphere", Report No. AFCRL-70-0134, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. 01731 (30 April 1970).

2. A. D. Pierce, W. A. Kinney, and C. Y. Kapper, "Atmosphere Acoustic Gravity Modes at Frequencies near and below Low Frequency Cutoff Imposed by Upper Boundary Conditions", Report No. AFCRL-TR-75-0639, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. 01731 (1 March 1976).

3. A. D. Pierce and W. A. Kinney, "Geometrical Acoustics Techniques in Far Field Infrasonic Waveform Syntheses", Report No. AFGL-TR-76-0055, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. 01731 (7 March 1976).

4. A. D. Pierce, C. A. Moo, and J. W. Posey, "Generation and Propagation of Infrasonic Waves", Report No. AFCRL-TR-73-0135, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. 01731 (30 April 1973).

5. J. E. Thomas, A. D. Pierce, E. A. Flinn, and L. B. Craine, "Bibliography on Infrasonic Waves", Geophys. J. Roy. Astron. Soc. 26, 399-426 (1971).

6. J. W. S. Rayleigh, "On the Vibrations of an Atmosphere", Phil. Mag. 29, 173-180 (1890).

7. H. Lamb, "On the Theory of Waves Propagated Vertically in the Atmosphere", Proc. London Math. Soc. 1, 122-141 (1908); "On Atmospheric Oscillations", Proc. Roy. Soc. London A84, 551-572 (1920).

8. G. I. Taylor "Waves and Tides in the Atmosphere", Proc. Roy. Soc. London A126, 169-183 (1929); "The Oscillations of the Atmosphere", Proc. Roy. Soc. London A156, 318-326 (1936).

9. C. L. Pekeris, "The Propagation of a Pulse in the Atmosphere", Proc. Roy. Soc. London A171, 434-449 (1939); "The Propagation of a Pulse in the Atmosphere, Part II", Phys. Rev. 73, 145-154 (1948).

10. R. S. Scoren, "The Dispersion of a Pressure Pulse in the Atmosphere", Proc. Roy. Soc. London A201, 137-157 (1950).

11. G. J. Symond, The Eruption of Krakatoa and Subsequent Phenomena (Trubner and Co., London, 1888).

12. F. J. W. Whipple, "On Phenomena Related to the Great Siberian Meteor", Quart. J. Roy. Meteor. Soc. 60, 505-513 (1934).

13. A. D. Pierce and C. A. Moo, "Theoretical Study of the Propagation of Infrasonic Waves in the Atmosphere", Report No. AFCRL-67-0172, Air Force Cambridge Research Laboratories, Hanscom AFB, Mass. 01731 (1967).

14. D. G. Harkrider, "Theoretical and Observed Acoustic-Gravity Waves from Explosive Sources in the Atmosphere", J. Geophys. Res. 69, 5295-5321 (1964).

15. A. D. Pierce and J. W. Posey, "Theory of the Excitation and Propagation of Lamb's Atmospheric Edge Mode from Nuclear Explosions", Geophys. J. Roy. Astron. Soc. 26, 341-368 (1971).

16. J. W. Posey, "Application of Lamb Edge Mode Theory in the Analysis of Explosively Generated Infrasound", Ph.D. Thesis, Dept. of Mech. Engrg., Mass. Inst. of Tech. (August, 1971).

17. N. A. Haskell, "Asymptotic Approximation for the Normal Modes in Sound Channel Wave Propagation", J. Appl. Phys. 22, 157-168 (1951).

18. F. Carlini, Ricerche sulla convergenza della serie che serva alla soluzione del problema di Keplero, Milan (1917).

19. G. Green, "On the Motion of Waves in a Variable Canal of Small Depth and Width", Trans. Camb. Phil. Soc. 6, 457-462 (1837).

20. C. Eckart, "Internal Waves in the Ocean", Phys. of Fluids 4, 791-799 (1961).

21. P. M. Morse and H. Feshbach, "Perturbation Methods for Scattering and Diffraction", Sec. 9.3 in Methods of Theoretical Physics, Vol. II (McGraw-Hill Book Co., New York, 1953) pp. 1092-1106.

22. A. D. Pierce, "Guided Infrasonic Modes in a Temperature and Wind-Stratified Atmosphere", J. Acoust. Soc. Amer. 41, 597-611 (1967).

23. W. L. Donn and D. M. Shaw, "Exploring the Atmosphere with Nuclear Explosions", Rev. of Geophys. 5, 53-82 (1967).

24. J. N. Brune, J. E. Nafe, and L. E. Alsop, "The Polar Phase Shift of Surface Waves on a Sphere", Bull. Seism. Soc. Amer. 51, 247-257 (1961).

25. P. W. Smith, "The Average Impulse Responses of a Shallow-Water Channel", J. Acoust. Soc. Amer. 50, 332-336 (1971); "Averaged Sound Transmission in Range-Dependent Channels", J. Acoust. Soc. Amer. 55, 1197-1204 (1974).

APPENDIX A

SOURCE DECK LISTING OF THE PRESENT VERSION OF INFRASONIC WAVEFORMS

This supercedes the source deck listing originally given by Pierce and Posey in AFCRL-70-0134. Changes incorporated include those described by Pierce, Moo, and Posey in AFCRL-TR-73-0135 and those described in the present report.

```
      PROGRAM INFRA(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,              MAIN
     1PUNCH,TAPE7=PUNCH,TAPE2)                                          MAIN
C     MAIN PROGRAM   .        7/23/69                                   MAIN
C                                                                       MAIN
C********************************************************************** MAIN
C                                                                       MAIN
C            PROGRAM TO SYNTHESIZE PRESSURE WAVEFORMS OF ACOUSTIC       MAIN
C            GRAVITY WAVES GENERATED BY NUCLEAR EXPLOSIONS IN THE       MAIN
C            ATMOSPHERE                                                 MAIN
C                                                                       MAIN
C********************************************************************** MAIN
C                                                                       MAIN
C                     ----ABSTRACT----                                  MAIN
C                                                                       MAIN
C TITLE - MAIN PROGRAM                                                  MAIN
C           GENERAL PURPOSE PROGRAM FOR STUDYING THE PROPAGATION OF NUCLEA MAIN
C           EXPLOSION GENERATED ACOUSTIC GRAVITY WAVES IN THE ATMOSPHERE.  MAIN
C                                                                       MAIN
C                THE ATMOSPHERE IS APPROXIMATED BY A MULTILAYER ATMOSPHER MAIN
C                WITH CONSTANT WIND VELOCITY AND TEMPERATURE IN EACH LAYE MAIN
C                THE NUMBER OF LAYERS, WIDTHS OF LAYERS, AND PROPERTIES O MAIN
C                LAYERS MAY BE SELECTED BY THE USER.  THE GROUND AT Z=0 I MAIN
C                ASSUMED FLAT AND RIGID.  THE UPPERMOST LAYER OF THE      MAIN
C                ATMOSPHERE IS ASSUMED TO BE UNBOUNDED FROM ABOVE.        MAIN
C                                                                       MAIN
C                THE SOURCE IS SPECIFIED BY ITS HEIGHT OF BURST AND ENERG MAIN
C                YIELD.  IT IS APPROXIMATED AS A POINT ENERGY SOURCE WITH MAIN
C                TIME DEPENDENCE CONFORMING TO CUBE ROOT (HYDRODYNAMIC)   MAIN
C                SCALING DERIVED FROM THE EFFECTS OF NUCLEAR WEAPONS      MAIN
C                (U.S. GOVERNMENT PRINTING OFFICE, 1962).                 MAIN
C                                                                       MAIN
C                THE OBSERVER LOCATION MAY BE SPECIFIED ARBITRARILY.      MAIN
C                HOWEVER, THE COMPUTATION INCLUDES ONLY CONTRIBUTIONS FRO MAIN
C                FULLY DUCTED GUIDED MODES AND ACCORDINGLY GIVES A SOLUTI MAIN
C                VALID (AT BEST) ONLY AT LARGE HORIZONTAL DISTANCES.      MAIN
C                ALSO, THE PROGRAMMING IS BASED ON THE PREMISE THAT ONLY  MAIN
C                PORTIONS OF MODES WITH PHASE VELOCITIES GREATER THAN THE MAIN
C                MAXIMUM WIND SPEED ARE TO BE INCLUDED INTO THE COMPUTATI MAIN
C                THE PROGRAM CANNOT THEREFORE BE APPLIED TO THE STUDY OF  MAIN
C                CRITICAL LAYER EFFECTS.                                  MAIN
C                                                                       MAIN
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)           MAIN
C                                                                       MAIN
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968               MAIN
C                                                                       MAIN
C                     ----USAGE----                                     MAIN
C                                                                       MAIN
C     ALL DATA IS INPUT IN THE NAMELIST FORMAT.  EACH SEQUENCE OF DATA  MAIN
C     MUST INCLUDE A NAM1 GROUP AT THE BEGINNING.                       MAIN
C                                                                       MAIN
C&NAM1      NSTART= , NPRNT= , NPNCH=       &END                        MAIN
C                                                                       MAIN
C     THE REMAINDER OF THE DATA TO BE SUPPLIED DEPENDS ON THE VALUE     MAIN
C     OF NSTART.                                                        MAIN
C                                                                       MAIN
C     ****NSTART=1****                                                  MAIN
C&NAM2      LANG'= , IMAX= , T= , , ,..., VKNTX= , , ,..., ETC.       & MAIN
C&NAM4      THETK= , V1= , V2= , OM1= , ETC.                          & MAIN
C&NAM6      ZSORCE= , ZOBS=     &END                                    MAIN
C&NAM9      YIELD=      &END                                            MAIN
C&NAM10     TFIRST= , TEND= , DELTT= , ROBS= , IOPT=       &END         MAIN
C                                                                       MAIN
C     ****NSTART=2****                                                  MAIN
C&NAM3      IMAX=  , CI= , ,..., VXI= , , ,..., ETC.       &END         MAIN
C&NAM4      THETK= , V1= , V2= , OM1= , ECT.           &END             MAIN
C&NAM6      ZSORCE= , ZOBS=     &END                                    MAIN
C&NAM9      YIELD=      &END                                            MAIN
C&NAM10     TFIRST= , TEND= , DELTT= , ROBS= , IOPT=       &END         MAIN
C                                                                       MAIN
C     ****NSTART=3****                                                  MAIN
```

```
C&NAM5     IMAX=    , CI=  ,   ...., VXI=   ,  ,  ...., ETC.     &END     MAIN
C&NAM6     ZSCRCE=  , ZOBS=    &END                                      MAIN
C&NAM8     YIELD=     &END                                               MAIN
C&NAM10    TFIRST=  , TEND=   , DELTT=   , ROBS=   , IOPT=     &END     MAIN
C                                                                        MAIN
C      ****NSTART=4****                                                  MAIN
C&NAM7     OMMOD=   ,   ,  ...., VPMOD=   ,   ...., MDFNO=   , ETC.      MAIN
C&NAM8     YIELD=     &END                                               MAIN
C&NAM10    TFIRST=  , TEND=   , DELTT=   , ROBS=   , IOPT=     &END     MAIN
C                                                                        MAIN
C      ****NSTART=5****                                                  MAIN
C&NAM9     MDFNO=   , KST=   ,   ...., KFIN=   ,   ...., OMMOD=   ,  , ETC.  MAIN
C&NAM10    TFIRST=  , TEND=   , DELTT=  ., ROBS=   , IOPT=     &END     MAIN
C                                                                        MAIN
C      ****NSTART=6****                                                  MAIN
C(NO ADDITIONAL DATA IS NEEDED.  COMPUTATION TERMINATES.)                MAIN
C                                                                        MAIN
C      FOR A COMPLETE LIST OF VARIABLES THAT ARE INCLUDED IN A GIVEN     MAIN
C      NAMELIST GROUP, SEE NAMELIST STATEMENTS IN PROGRAM.  NOTE THAT    MAIN
C      DATA INPUT BY READ(5,NAM1), READ(5,NAM2), ETC., NEED NOT INCLUDE  MAIN
C      VALUES OF ALL VARIABLES IN THE CORRESPONDING NAMELIST GROUP.  ONE MAIN
C      NEED ONLY INPUT THOSE VALUES NEEDED FOR THE CALCULATION AND WHICH MAIN
C      ARE NOT ALREADY IN STORAGE.                                       MAIN
C                                                                        MAIN
C      DATA ASSOCIATED WITH NAM3, NAM5, NAM7, AND NAM9 SHOULD IN GENERAL MAIN
C      NOT BE SUPPLIED ARBITRARILY, BUT MAY BE OBTAINED FROM PREVIOUS    MAIN
C      RUNS OF THE PROGRAM.  IF NSTART=1, NPNCH=1, DATA CARDS FOR NAM3,  MAIN
C      NAM5, NAM7, AND NAM9 ARE AUTOMATICALLY PUNCHED.  IF NSTART=2,     MAIN
C      NPNCH=1, DATA CARDS FOR NAM5, NAM7, AND NAM9 ARE PUNCHED.  IF     MAIN
C      NSTART=3, NPNCH=1, DATA CARDS FOR NAM7 AND NAM9 ARE PUNCHED.  IF  MAIN
C      NSTART=4, NPNCH=1, DATA CARDS FOR NAM9 ARE PUNCHED.               MAIN
C                                                                        MAIN
C      THE NEXT BATCH OF DATA AFTER NAM10 SHOULD BE NAM1.  THE LAST DATA MAIN
C      CARD SHOULD BE NAM1 WITH NSTART=6.                                MAIN
C                                                                        MAIN
C                  ----EXTERNAL SUBROUTINES REQUIRED----                 MAIN
C                                                                        MAIN
C      SUBROUTINE      TYPE      CALLED BY                               MAIN
C                                                                        MAIN
C      AAAA            SUB       ELINT,MMMM,NAMPDE,NMOFN                 MAIN
C      AKI             SUB       TMPT                                    MAIN
C      ALLMOD          SUB       MAIN                                    MAIN
C      AMBNT           SUB       PAMPDE                                  MAIN
C      ATMOS           SUB       MAIN                                    MAIN
C      AXIS1           SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      BBBB            SUB       ELINT                                   MAIN
C      CAI             FUNC      BBBB,MMMM                               MAIN
C      DXDY1           SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      ELINT           SUB       TOTINT                                  MAIN
C      ENDPLT          SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      FNMOD1          FUNC      MODETR (EXTERNAL FOR ARG. OF RTMI)      MAIN
C      FNMOD2          FUNC      MODETR (EXTERNAL FOR ARG. OF RTMI)      MAIN
C      LNGTHN          SUB       TABLE                                   MAIN
C      MMMM            SUB       NAMPDE,RRRR                             MAIN
C      MODETR          SUB       ALLMOD                                  MAIN
C      MODLST          SUB       MAIN                                    MAIN
C      MPOUT           SUB       TABLE                                   MAIN
C      NAMPDE          SUB       PAMPDE                                  MAIN
C      NEWPLT          SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      NMOFN           SUB       FNMOD1,FNMOD2,LNGTHN,MPOUT,WIDEN        MAIN
C      NUMBR1          SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      NXMODE          SUB       ALLMOD                                  MAIN
C      NXPNT           SUB       MODETR                                  MAIN
C      PAMPDE          SUB       MAIN                                    MAIN
C      PHASE           SUB       SOURCE                                  MAIN
C      PLOT1           SUB       TMPT  (M.I.T. CALCOMP ROUTINE)          MAIN
C      PPAMP           SUB       MAIN                                    MAIN
C      PRATMO          SUB       MAIN                                    MAIN
C      RRRR            SUB       NMOFN                                   MAIN
C      RTMI            SUB       MODETR (IBM SCIENTIFIC SUBROUTINE)      MAIN
```

```
C       SAI              FUNC      ROBB,MMMM                                    MAIN
C       SCLGPH           SUB       THPT  (M.I.T. CALCOMP ROUTINE)               MAIN
C       SOURCE           SUB       PPAMP                                        MAIN
C       SUSPCT           SUB       TABLE                                        MAIN
C       SYMBOLS          SUB       THPT  (M.I.T. CALCOMP ROUTINE)               MAIN
C       TABLE            SUB       MAIN                                         MAIN
C       TABPRT           SUB       MAIN                                         MAIN
C       THPT             SUB       MAIN                                         MAIN
C       TOTINT           SUB       NAMPOE                                       MAIN
C       UPINT            SUB       TOTINT                                       MAIN
C       USEAS            SUB       TOTINT                                       MAIN
C       WIDEN            SUB       TABLE                                        MAIN
C                                                                               MAIN
C                      ----INPUTS THROUGH NAMELIST READ STATEMENTS----          MAIN
C                                                                               MAIN
C NAM1 -- NAMELIST GROUP 1                                                      MAIN
C                                                                               MAIN
C       NSTART       =FLAG DENOTING POINT IN MAIN PROGRAM AT WHICH COMPUTA MAIN
C                     TION BEGINS.  POSSIBLE VALUES OF 1 THROUGH 5 CAUSE        MAIN
C                     NAM2, NAM3, NAM5, NAM7, OR NAM9 TO BE READ.  NSTART= MAIN
C                     CAUSES TERMINATION OF PROGRAM EXECUTION.                  MAIN
C       NPRNT        =FLAG FOR PRINTING OPTION.  IF NPRNT .LE. 0, A MINIMA MAIN
C                     AMOUNT OF PRINTOUT WILL BE RETURNED.                      MAIN
C       NPNCH        =FLAG FOR PUNCHING OPTION.  IF NPNCH .LE. 0, NO INFO  MAIN
C                     WILL BE PUNCHED ON CARDS.                                 MAIN
C                                                                               MAIN
C NAM2 -- NAMELIST GROUP 2                                                      MAIN
C                                                                               MAIN
C       LANGLE       =INTEGER WHICH SPECIFIES WHICH TYPE OF ATMOSPHERIC DA MAIN
C                     IS INPUT.  IF LANGEL .LE. 0, THE WIND COMPONENTS IN  MAIN
C                     KNOTS ARE SPECIFIED, WHILE IF LANGE .GT. 0, THE WIND MAIN
C                     MAGNITUDE AND DIRECTION ARE SPECIFIED FOR EACH LAYER MAIN
C       IMAX         =NUMBER OF LAYERS OF FINITE THICKNESS IN MULTILAYER   MAIN
C                     ATMOSPHERE.                                               MAIN
C       T(I)         =TEMPERATURE IN DEGREES KELVIN IN THE I-TH LAYER.     MAIN
C       VKNTX(I)     =X (WEST TO EAST) COMPONENT OF WIND VELOCITY IN I-TH  MAIN
C                     LAYER.                                                    MAIN
C       VKNTY(I)     =Y (SOUTH TO NORTH) COMPONENT OF WIND VELOCITY IN I-T MAIN
C                     LAYER.                                                    MAIN
C       WINDV(I)     =WIND VELOCITY MAGNITUDE IN KNOTS IN I-TH LAYER.      MAIN
C       WANGLE(I)    =WIND VELOCITY DIRECTION IN DEGREES, RECKONED COUNTER MAIN
C                     CLOCKWISE FROM X-AXIS.                                    MAIN
C       ZI(I)        =HEIGHT IN KILOMETERS OF THE TOP OF THE I-TH LAYER OF MAIN
C                     FINITE THICKNESS.                                         MAIN
C                                                                               MAIN
C NAM3 -- NAMELIST GROUP 3                                                      MAIN
C                                                                               MAIN
C       IMAX         =NUMBER OF LAYERS OF FINITE THICKNESS.                     MAIN
C       CI(I)        =SOUND SPEED IN KM/SEC IN I-TH LAYER.                      MAIN
C       VXI(I)       =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC). MAIN
C       VYI(I)       =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC). MAIN
C       HI(I)        =THICKNESS IN KM OF I-TH LAYER OF FINITE THICKNESS.   MAIN
C                                                                               MAIN
C NAM4 -- NAMELIST GROUP 4                                                      MAIN
C                                                                               MAIN
C       THETKO       =DIRECTION IN DEGREES TO OBSERVER, RECKONED COUNTER   MAIN
C                     CLOCKWISE FROM X AXIS.                                    MAIN
C       V1           =LOWER BOUND IN KM/SEC OF PHASE VELOCITY INTERVAL CON MAIN
C                     SIDERED FOR NORMAL MODE TABULATION                        MAIN
C       V2           =UPPER BOUND IN KM/SEC OF PHASE VELOCITY INTERVAL CON MAIN
C                     SIDERED FOR NORMAL MODE TABULATION                        MAIN
C       OM1          =MINIMUM ANGULAR FREQUENCY IN RAD/SEC CONSIDERED FOR  MAIN
C                     NORMAL MODE TABULATION.                                   MAIN
C       OM2          =MAXIMUM ANGULAR FREQUENCY IN RAD/SEC CONSIDERED FOR  MAIN
```

```
C                   NORMAL MODE TABULATION.                           MAIN
C       NOMI       =INITIAL NUMBER OF DISCRETE FREQUENCIES BETWEEN OM1 MAIN
C                   AND OM2, INCLUSIVE, AT WHICH NORMAL MODE DISPERSION MAIN
C                   FUNCTION IS STUDIED.                              MAIN
C       NVPI       =INITIAL NUMBER OF DISCRETE PHASE VELOCITIES BETWEEN MAIN
C                   V1 AND V2, INCLUSIVE, AT WHICH NORMAL MODE DISPERSIO MAIN
C                   FUNCTION IS STUDIED.                              MAIN
C       MAXMOD     =MAXIMUM NUMBER OF MODES TO BE TABULATED.          MAIN
C                                                                     MAIN
C NAM5 -- NAMELIST GROUP 5                                            MAIN
C                                                                     MAIN
C       IMAX       =NUMBER OF LAYERS OF FINITE THICKNESS              MAIN
C       CI(I)      =SOUND SPEED IN KM/SEC IN I-TH LAYER               MAIN
C       VXI(I)     =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC) MAIN
C       VYI(I)     =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC) MAIN
C       HI(I)      =THICKNESS IN KM OF I-TH LAYER OF FINITE THICKNESS MAIN
C       THETKO     =DIRECTION IN DEGREES TO OBSERVER, RECKONED COUNTER MAIN
C                   CLOCKWISE FROM X AXIS                             MAIN
C       MOFNO      =NUMBER OF NORMAL MODES FOUND                      MAIN
C       KST(N)     =INDEX OF FIRST TABULATED POINT IN N-TH MODE       MAIN
C       KFIN(N)    =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN   MAIN
C                   GENERAL, KFIN(N)=KST(N+1)-1.                      MAIN
C       OMMOD(N)   =ARRAY STORING ANGULAR FREQUENCY ORDINATE (RAD/SEC) O MAIN
C                   POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                   FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).         MAIN
C       VPMOD(N)   =ARRAY STORING PHASE VELOCITY ORDINATE (KM/SEC) OF MAIN
C                   POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                   FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).         MAIN
C                                                                     MAIN
C NAM6 -- NAMELIST GROUP 6                                            MAIN
C                                                                     MAIN
C       ZSCRCE     =HEIGHT IN KM OF BURST ABOVE GROUND                MAIN
C       ZOBS       =HEIGHT IN KM OF OBSERVER ABOVE GROUND             MAIN
C                                                                     MAIN
C NAM7 -- NAMELIST GROUP 7                                            MAIN
C                                                                     MAIN
C       OMMOD(N)   =ARRAY STORING ANGULAR FREQUENCY ORDINATE (RAD/SEC) O MAIN
C                   POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                   FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).         MAIN
C       VPMOD(N)   =ARRAY STORING PHASE VELOCITY ORDINATE (KM/SEC) OF MAIN
C                   POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                   FOR N BETWEEN KST(NMODE) AND KFIN(NMODE)          MAIN
C       MOFNO      =NUMBER OF NORMAL MODES FOUND                      MAIN
C       KST(N)     =INDEX OF FIRST TABULATED POINT IN N-TH MODE       MAIN
C       KFIN(N)    =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN   MAIN
C                   GENERAL, KFIN(N)=KST(N+1)-1.                      MAIN
C       AMP(J)     =AMPLITUDE FACTOR FOR GUIDED WAVE EXCITED BY POINT MAIN
C                   ENERGY SOURCE.  UNITS ARE KM**(-1).  THE J-TH ELEMEN MAIN
C                   CORRESPONDS TO ANGULAR FREQUENCY OMMOD(J) AND PHASE MAIN
C                   VELOCITY VPMOD(J).  THE AMPLITUDE FACTOR IS APPROPRI MAIN
C                   TO THE NMODE-TH MODE IF J .GE. KST(NMODE) AND J .LE. MAIN
C                   KFIN(NMODE).  A DETAILED DEFINITION OF AMP(J) IS GIV MAIN
C                   IN THE LISTING OF SUBROUTINE NAMPDE.              MAIN
C       ALAM       =A SCALING FACTOR DEPENDENT ON HEIGHT OF BURST. EQUAL MAIN
C                   TO CUBE ROOT OF (PRESSURE AT GROUND)/(PRESSURE AT MAIN
C                   BURST HEIGHT) TIMES (SOUND SPEED AT GROUND)/(SOUND MAIN
C                   SPEED AT BURST HEIGHT).  SEE SUBROUTINE PAMPDE.   MAIN
C       FACT       =A GENERAL AMPLITUDE FACTOR DEPENDENT ON BURST HEIGHT MAIN
C                   AND OBSERVER HEIGHT.  A PRECISE DEFINITION IS GIVEN MAIN
C                   IN THE LISTING OF SUBROUTINE PAMPDE.              MAIN
C                                                                     MAIN
C NAM8 -- NAMELIST GROUP 8                                            MAIN
C                                                                     MAIN
C       YIELD      =ENERGY YIELD OF EXPLOSION IN EQUIVALENT KILOTONS (KT MAIN
```

```
C                   OF TNT.  1 KT = 4.2X(10)**19 ERGS.                      MAIN
C                                                                           MAIN
C NAM9 -- NAMELIST GROUP 9                                                  MAIN
C                                                                           MAIN
C       MOFNO       =NUMBER OF NORMAL MODES FOUND                           MAIN
C       KST(N)      =INDEX OF FIRST TABULATED POINT IN N-TH MODE            MAIN
C       KFIN(N)     =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN        MAIN
C                    GENERAL, KFIN(N)=KST(N+1)-1                            MAIN
C       OMMOD(N)    =ARRAY STORING ANGULAR FREQUENCY ORDINATE (RAD/SEC) O MAIN
C                    POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                    FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).              MAIN
C       VPMOD(N)    =ARRAY STORING PHASE VELOCITY ORDINATE (KM/SEC) OF      MAIN
C                    POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR MAIN
C                    FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).              MAIN
C       AMPLTO(N)   =AMPLITUDE FACTOR REPRESENTING TOTAL MAGNITUDE OF       MAIN
C                    FOURIER TRANSFORM OF WAVEFORM CONTRIBUTION OF SINGLE MAIN
C                    GUIDED MODE AT FREQUENCY OMMOD(N).  IT REPRESENTS TH MAIN
C                    AMPLITUDE OF NMODE-TH MODE IF N IS BETWEEN KST(NMODE MAIN
C                    AND KFIN(NMODE), INCLUSIVE.  FOR PRECISE DEFINITION, MAIN
C                    SEE SUBROUTINE PPAMP.                                  MAIN
C       PHASQ(N)    =PHASE LAG AT FREQUENCY OMMOD(N) FOR NMODE MODE WHEN    MAIN
C                    N BETWEEN KST(NMODE) AND KFIN(NMODE), RESPECTIVELY.    MAIN
C                    THE INTEGRAND IS UNDERSTOOD TO HAVE THE FORM           MAIN
C                    AMPLTO*COS(OMMOD*(TIME-DISTANCE/VPMOD)+PHASQ).  FOR    MAIN
C                    PRECISE DEFINITION OF PHASQ,  SEE SUBROUTINES TMPT     MAIN
C                    AND PPAMP.                                             MAIN
C                                                                           MAIN
C NAM10 -- NAMELIST GROUP 10                                                MAIN
C                                                                           MAIN
C       TFIRST      =FIRST TIME RELATIVE TO TIME OF DETONATION FOR WHICH    MAIN
C                    WAVEFORM IS COMPUTED.  UNITS ARE IN SECONDS.           MAIN
C       TEND        =APPROXIMATE TIME VALUE CORRESPONDING TO LAST POINT     MAIN
C                    TABULATED FOR WAVEFORM (RELATIVE TO TIME OF DETONATI MAIN
C                    FOR PRECISE DEFINITION, SEE SUBROUTINE TMPT.           MAIN
C       DELTT       =INCREMENT OF TIME VALUES IN SECONDS FOR WHICH SUCCES MAIN
C                    SIVE WAVEFORM POINTS ARE TABULATED.                    MAIN
C       ROBS        =MAGNITUDE OF HORIZONTAL DISTANCE IN KM BETWEEN SOURC MAIN
C                    AND OBSERVER.                                          MAIN
C       IOPT        =INTEGER CONTROLLING WHICH MODES ARE INCLUDED IN THE    MAIN
C                    COMPUTED WAVEFORM.  FOR PRECISE DEFINITION, SEE        MAIN
C                    SUBROUTINE TMPT.                                       MAIN
C                                                                           MAIN
C                                                                           MAIN
C                    ----PROGRAM FOLLOWS BELOWS ----                        MAIN
C                                                                           MAIN
C                                                                           MAIN
C DIMENSION STATEMENTS                                                      MAIN
      DIMENSION CI(100),VXI(100),VYI(100),HI(100),AMP(1000),AMPLTO(1000 MAIN
      DIMENSION T(100),VKNTX(100),VKNTY(100),ZI(100),PHASQ(1000)            MAIN
      DIMENSION WANGLE(100),WINDY(100)                                      MAIN
      DIMENSION OM(100),VP(100),INMODE(10000)                               MAIN
      DIMENSION KST(10),KFIN(10),OMMOD(1000),                               MAIN
     1VPMOD(1000),AKI(1000),IBUF(1400)                                      MAIN
      DIMENSION OMGR1(50),VPGR1(50),AKIGR1(50),OMGR0(50),                   MAIN
     1VPGR0(50),AKIGR0(50),AMPGR0(50),AMPGR1(50)                            MAIN
C                                                                           MAIN
C ALOCATION OF VARIABLES TO COMMON STORAGE                                  MAIN
      COMMON IMAX,CI,VXI,VYI,HI                                             MAIN
C                                                                           MAIN
C NAMELIST STATEMENTS                                                       MAIN
      NAMELIST /NAM1/ NSTART,NPRNT,NPNCH,NCMPL                              MAIN
      NAMELIST /NAM2/ LANGLE,IMAX,T,VKNTX,VKNTY,WINDY,WANGLE,ZI             MAIN
      NAMELIST /NAM3/ IMAX,CI,VXI,VYI,HI                                    MAIN
      NAMELIST /NAM4/ THETKO,V1,V2,OM1,OM2,NOMI,NVPI,MAXMOD                 MAIN
```

```
      NAMELIST /NAM5/ IMAX,CI,VXI,VYI,HI,THETKO,MOFNO,KST,KFIN,OMMOD,      MAIN
     1 VPMOD                                                               MAIN
      NAMELIST /NAM6/ ZSCRCE,ZOBS                                          MAIN
      NAMELIST /NAM7/ CMMOD,VPMOD,MOFNO,KST,KFIN,AMP,ALAM,FACT             MAIN
      NAMELIST /NAM8/ YIELD                                                MAIN
      NAMELIST /NAM9/ MOFNO,KST,KFIN,OMMOD,VPMOD,AMPLTD,PHASQ              MAIN
      NAMELIST /NAM10/ TFIRST,TEND,DELTT,ROBS,IOPT                         MAIN
      NAMELIST /NAM51/ MNGR1,NPGR1,MNGR0,NPGR0,OMGR1,VPGR1,OMGR0,          MAIN
     1VPGR0,AKIGR1,AKIGR0,AMPGR0,AMPGR1                                    MAIN
C                                                                          MAIN
C                                                                          MAIN
C BEFORE ANY DATA IS READ IN, ALL NAMELIST VALUES ARE PRESET TO ZERO.      MAIN
C THIS IS DONE SIMPLY TO MAKE NAMELIST PRINTOUT EASIER TO READ.            MAIN
      NSTART=0                                                             MAIN
      NPRNT=0                                                              MAIN
      NPNCH=0                                                              MAIN
      NCMPL=0                                                              MAIN
      LANGLE=0                                                             MAIN
      IMAX=0                                                               MAIN
      THETKO=0.0                                                           MAIN
      V1=0.0                                                               MAIN
      V2=0.0                                                               MAIN
      OM1=0.0                                                              MAIN
      OM2=0.0                                                              MAIN
      NOMI=0                                                               MAIN
      NVPI=0                                                               MAIN
      MAXMOD=0                                                             MAIN
      MOFNO=0                                                              MAIN
      ZSCRCE=0.0                                                           MAIN
      ZOBS=0.0                                                             MAIN
      ALAM=0.0                                                             MAIN
      FACT=0.0                                                             MAIN
      YIELD=0.0                                                            MAIN
      TFIRST=0.0                                                           MAIN
      TEND=0.0                                                             MAIN
      DELTT=0.0                                                            MAIN
      ROBS=0.0                                                             MAIN
      IOPT=0                                                               MAIN
      DO 21 IPR=1,100                                                      MAIN
      CI(IPR)=0.0                                                          MAIN
      VXI(IPR)=0.0                                                         MAIN
      VYI(IPR)=0.0                                                         MAIN
      HI(IPR)=0.0                                                          MAIN
      T(IPR)=0.0                                                           MAIN
      VKNTX(IPR)=0.0                                                       MAIN
      VKNTY(IPR)=0.0                                                       MAIN
      ZI(IPR)=0.0                                                          MAIN
      WANGLE(IPR)=0.0                                                      MAIN
      WINDY(IPR)=0.0                                                       MAIN
      OM(IPR)=0.0                                                          MAIN
   21 VP(IPR)=0.0                                                          MAIN
      DO 31 IPR=1,10                                                       MAIN
      KST(IPR)=0                                                           MAIN
   31 KFIN(IPR)=0                                                          MAIN
      DO 41 IPR=1,1000                                                     MAIN
      AMP(IPR)=0.0                                                         MAIN
      AMPLTD(IPR)=0.0                                                      MAIN
      PHASQ(IPR)=0.0                                                       MAIN
      OMMOD(IPR)=0.0                                                       MAIN
      AKI(IPR)=0.0                                                         MAIN
   41 VPMOD(IPR)=0.0                                                       MAIN
C                                                                          MAIN
C                                                                          MAIN
C START OF EXECUTABLE PORTION OF PROGRAM
```

```
C
C NEWPLT IS A CALCOMP SUBROUTINE WHICH INITIATES THE CALCOMP PLOTTER
C TAPE FILE.  5640 IS THE M.I.T. COMPUTATION CENTER PROBLEM NO.  5923 I
C THE PROGRAMMER NO.  GRAPH PAPER WITH BLACK INK IS REQUESTED.
      CALL PLOTS(IBUF,1400,2,00)
C
    1 READ (5,NAM1)
C
C IT IS CONSIDERED GOOD PRACTICE TO HAVE INPUT DATA PRINTED ON OUTPUT
      WRITE (6,37)
   37 FORMAT(1H ///// 27H1NAM1 HAS JUST BEEN READ IN)
      WRITE (6,NAM1)
C
C CURRENT VALUE OF NSTART CONTROLS THE STAGE AT WHICH COMPUTATION BEGIN
C SINCE COMPUTED GO TO STATEMENTS SOMETIMES DO NOT COMPILE CORRECTLY IF
C INDEX IS NOT EXPLICITLY DEFINED, WE PLAY IT SAFE WITH REDUNDANT
C STATEMENT.
      NSTART=NSTART
C
      GO TO (200,300,400,500,600,999),NSTART
C
C WE ARRIVE HERE IF NSTART=1
  200 READ (5,NAM2)
C
      WRITE (6,237)
  237 FORMAT(1H ///// 27H NAM2 HAS JUST BEEN READ IN)
      WRITE (6,NAM2)
C
C CONVERT ATMOSPHERIC DATA TO STANDARD FORM
      CALL ATMOS(T,VKNTX,VKNTY,ZI,WANGLE,WINDY,LANGLE)
      IF( NPRNT .LE. 0) GO TO 270
C
C PRINT ATMOSPHERIC PROFILE IF NPRNT .GT. 0
      CALL PRATMO
C
  270 IF( NPNCH .LE. 0) GO TO 305
C
C PUNCH NAM3 DATA IF NPNCH .GT. 0
      WRITE (7,271)
  271 FORMAT ( 7H &NAM3 )
      IUHS = IMAX + 1
      WRITE (7,272) IMAX,(CI(I),I=1,IUHS)
  272 FORMAT (            10H   IMAX = ,I3,1H,   / 9H   CI = /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,274) (VXI(I),I=1,IUHS)
  274 FORMAT(9H   VXI = /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,276) (VYI(I),I=1,IUHS)
  276 FORMAT(9H   VYI = /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,278) ( HI(I),I=1,IUHS)
  278 FORMAT ( 8H   HI = /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE (7,279)
  279 FORMAT ( 6H &END  )
      WRITE (6,583)
      WRITE (6,271)
      WRITE (6,272) IMAX,(CI(I),I=1,IUHS)
      WRITE(6,274) (VXI(I),I=1,IUHS)
      WRITE(6,276) (VYI(I),I=1,IUHS)
      WRITE(6,278) ( HI(I),I=1,IUHS)
      WRITE (6,279)
  280 GO TO 305
C
```

```
C WE ARRIVE HERE IF NSTART=2                                          MAIN
  300 READ (5,NAM3)                                                   MAIN
      WRITE (6,302)                                                   MAIN
  302 FORMAT(1H ///// 27H NAM3 HAS JUST BEEN READ IN)                 MAIN
      WRITE (6,NAM3)                                                  MAIN
      IF( NPRNT .LE. 0) GO TO 305                                     MAIN
C PRINT ATMOSPHERIC PROFILE IF NPRNT .GT. 0                            MAIN
      CALL PPATMO                                                     MAIN
C                                                                     MAIN
C CONTINUING FROM 270, 280, 302, OR 303                               MAIN
  305 READ (5,NAM4)                                                   MAIN
      WRITE (6,307)                                                   MAIN
  307 FORMAT(1H ///// 27H NAM4 HAS JUST BEEN READ IN)                 MAIN
      WRITE (6,NAM4)                                                  MAIN
C                                                                     MAIN
C CONVERT THETKD FROM DEGREES TO RADIANS                              MAIN
      THETK = (3.14159) * THETKD / 180.0                              MAIN
      NOM = NOMI                                                      MAIN
      NVP = NVPI                                                      MAIN
C                                                                     MAIN
C CONSTRUCT TABLE OF INMODE VALUES                                    MAIN
      CALL TABLE(OM1,OM2,V1,V2,NOM,NVP,THETK,OM,VP,INMODE,NPRNT)      MAIN
C                                                                     MAIN
C COMPUTE DISPERSION CURVES OF GUIDED MODES                           MAIN
      CALL ALLMOD(NVP,NOM,MAXMCO,MOFND,OM,VP,KST,KFIN,OMMOD,VPMOD,    MAIN
     1  INMODE,THETK,KWOP)                                            MAIN
      IF(NCHPL .LE. 0) GO TO 309                                      MAIN
      READ(5,NAM51)                                                   MAIN
      KBEGIN = KST(MNGR1)                                             MAIN
      KENDI = KFIN(MNGR0)                                             MAIN
      KENDF = KBEGIN + NPGR0 + NPGR1 - 1                              MAIN
      IF(KENDF .LE. KENDI) GO TO 3085                                 MAIN
      KFINP1 = KENDI + 1                                              MAIN
      KFINMD = KFIN(MOFND)                                            MAIN
      DO 3081 LL = KFINP1,KFINMD                                      MAIN
      L = KFINMD + KFINP1 - LL                                        MAIN
      LNEW = L + KENDF - KENDI                                        MAIN
      OMMOD(LNEW) = OMMOD(L)                                          MAIN
      VPMOD(LNEW) = VPMOD(L)                                          MAIN
      AKI(LNEW) = AKI(L)                                              MAIN
      AMP(LNEW) = AMP(L)                                              MAIN
 3081 CONTINUE                                                        MAIN
      MNGROP1 = MNGR0 + 1                                             MAIN
      DO 3082 KKL = MNGROP1,MOFND                                     MAIN
      KL = MNGROP1 + MOFND - KKL                                      MAIN
      KFIN(KL) = KFIN(KL) + KENDF - KENDI                             MAIN
 3082 KST(KL) = KST(KL) + KENDF - KENDI                               MAIN
      GO TO 3088                                                      MAIN
 3085 CONTINUE                                                        MAIN
      KFINP1 = KFIN(MNGR0) + 1                                        MAIN
      KFINMD = KFIN(MOFND)                                            MAIN
      DO 3086 L = KFINP1,KFINMD                                       MAIN
      LNEW = L + KENDF - KENDI                                        MAIN
      OMMOD(LNEW) = OMMOD(L)                                          MAIN
      VPMOD(LNEW) = VPMOD(L)                                          MAIN
      AKI(LNEW) = AKI(L)                                              MAIN
      AMP(LNEW) = AMP(L)                                              MAIN
 3086 CONTINUE                                                        MAIN
      MNGROP1 = MNGR0 + 1                                             MAIN
      DO 3087 KKL = MNGROP1,MOFND                                     MAIN
      KL = MNGROP1 + MOFND - KKL                                      MAIN
      KFIN(KL) = KFIN(KL) + KENDF - KENDI                             MAIN
 3087 KST(KL) = KST(KL) + KENDF - KENDI                               MAIN
 3088 CONTINUE                                                        MAIN
```

```
      KST(MNGR1) = KBEGIN
      KFIN(MNGR1) = KST(MNGR1) + NPGR1 - 1
      KST(MNGR0) = KFIN(MNGR1) + 1
      KFIN(MNGR0) = KST(MNGR0) + NPGR0 - 1
      DO 3188 L = 1,NPGR1
      LNEW = KST(MNGR1) + L - 1
      OMMOD(LNEW) = OMGR1(L)
      VPMOD(LNEW) = VPGR1(L)
      AKI(LNEW) = AKIGR1(L)
      AMP(LNEW) = AMPGR1(L)
 3188 CONTINUE
      DO 3089 L = 1,NPGR0
      LNEW = KST(MNGR0) + L - 1
      OMMOD(LNEW) = OMGR0(L)
      VPMOD(LNEW) = VPGR0(L)
      AKI(LNEW) = AKIGR0(L)
      AMP(LNEW) = AMPGR0(L)
 3089 CONTINUE
  309 CONTINUE
C
C CHECK TO SEE IF ANY MODES WERE FOUND
      IF( KWOP .GE. 0) GO TO 320
C
C EXIT IF KWOP .LT. 0
      WRITE (6,311) KWOP
  311 FORMAT(1H , 5HKWOP=, I3)
      CALL EXIT
C
C CONTINUING WITH KWOP .GE. 0 FROM 308
  IF (NPRNT .LE. 0) GO TO 350
C PRINT NORMAL MODE DISPERSION CURVES
      CALL MODLST(MOFND,OMMOD,VPMOD,AKI,KST,KFIN)
C
C CONTINUING FROM 320 OR 321
  350 IF( NPNCH .LE. 0) GO TO 360
C
C PUNCH NAM5 DATA IF NPNCH .GT. 0
      WRITE (7,351)
  351 FORMAT ( 7H &NAM5 )
      IUHS = IMAX + 1
      WRITE (7,272) IMAX,(CI(I),I=1,IUHS)
      WRITE(7,274) (VXI(I),I=1,IUHS)
      WRITE(7,276) (VYI(I),I=1,IUHS)
      WRITE(7,279) ( HI(I),I=1,IUHS)
      WRITE (7,352) THETKO,MOFND,(KST(I),I=1,MOFND)
  352 FORMAT (11H   THETKO =,G16.8,1H,/10H   MOFND =,I3,1H,/8H   KST =/
   1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,355) (KFIN(I),I=1,MOFND)
  355 FORMAT ( 10H   KFIN = /
   1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      KLAST = KFIN(MOFND)
      WRITE (7,357)  (OMMOD(I),I=1,KLAST)
  357 FORMAT ( 11H   OMMOD = /
   1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,359) (VPMOD(I),I=1,KLAST)
  359 FORMAT ( 11H   VPMOD = /
   1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE (7,279)
      WRITE (6,583)
      WRITE (6,351)
      WRITE (6,272) IMAX,(CI(I),I=1,IUHS)
      WRITE(6,274) (VXI(I),I=1,IUHS)
      WRITE(6,276) (VYI(I),I=1,IUHS)
      WRITE(6,279) ( HI(I),I=1,IUHS)
```

```
      WRITE (6,352) THETKD,MOFNO,(KST(I),I=1,MOFNO)
      WRITE(6,355) (KFIN(I),I=1,MOFNO)
      WRITE (6,357)  (OMMOD(I),I=1,KLAST)
      WRITE(6,359) (VPMOD(I),I=1,KLAST)
      WRITE (6,279)
C
C CONTINUING FROM 350 OR 351
  360 GO TO 415
C
C
C WE ARRIVE HERE IF KSTART=3
  400 READ (5,NAM5)
      WRITE (6,403)
  403 FORMAT(1H ///// 27H NAM5 HAS JUST BEEN READ IN)
      WRITE (6,NAM5)
C
C CONVERT THETKD FROM DEGREES TO RADIANS
      THETK = (3.14159) * THETKD / 180.0
C
C CONTINUING FROM 360 OR 402
  415 READ (5,NAM6)
      WRITE (6,417)
  417 FORMAT(1H ///// 27H NAM6 HAS JUST BEEN READ IN)
      WRITE (6,NAM6)
C
C COMPUTE YIELD INDEPENDENT AMPLITUDE FACTORS FOR GUIDED MODES
      CALL PAMFDE(ZSCRCE,ZOBS,MOFNO,KST,KFIN,OMMOD,VPMOD,AKI,
     1AMP,ALAM,FACT,THETK,NPRNT)
C
  450 IF( NPNCH .LE. 0) GO TO 460
C
C PUNCH NAM7 DATA IF NPNCH .GT. 0
      KLAST = KFIN(MOFNO)
      WRITE (7,451)(AMP(I),I=1,KLAST)
  451 FORMAT ( 7H &NAM7 / 9H   AMP =./
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE (7,452) ALAM,FACT
  452 FORMAT ( 10H   ALAM = ,G16.8,1H, / 10H   FACT = ,G16.8,1H, )
      WRITE (7,455) MOFNO,(KST(I),I=1,MOFNO)
  455 FORMAT (                                10H   MOFNO =,I3,1H,/8H   KST =/
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE(7,355) (KFIN(I),I=1,MOFNO)
      WRITE (7,357)  (OMMOD(I),I=1,KLAST)
      WRITE(7,359) (VPMOD(I),I=1,KLAST)
      WRITE (7,279)
      WRITE (6,583)
      WRITE (6,451)(AMP(I),I=1,KLAST)
      WRITE (6,452) ALAM,FACT
      WRITE (6,455) MOFNO,(KST(I),I=1,MOFNO)
      WRITE(6,355) (KFIN(I),I=1,MOFNO)
      WRITE (6,357)  (OMMOD(I),I=1,KLAST)
      WRITE(6,359) (VPMOD(I),I=1,KLAST)
  459 WRITE (6,279)
C
C CONTINUING FROM 450 OR 459
  460 GO TO 515
C
C
C WE ARRIVE HERE IF KSTART=4
  500 READ (5,NAM7)
      WRITE (6,501)
  501 FORMAT(1H /////  27H NAM7 HAS JUST BEEN READ IN)
  502 WRITE (6,NAM7)
C
```

```
C CONTINUING FROM 460 OR 502
  515 READ (5,NAM8)
      WRITE (6,516)
  516 FORMAT( 1H ///// 27H NAM8 HAS JUST BEEN READ IN)
  517 WRITE (6,NAM8)
C
C COMPUTE YIELD DEPENDENT AMPLITUDES AND PHASE TERMS OF GUIDED MODES
      CALL PPAMP(YIELD,MOFNO,KST,KFIN,OMMOD,VPMOD,
     1AMP,ALAM,FACT,AMPLTD,PHASQ)
  518 IF( NPRNT .LE. 0 ) GO TO 580
C THE RESULTS OF CALLING PPAMP ARE PRINTED OUT BY CALLING TABPRT
  CALL TABPRT(YIELD,MOFNO,KST,KFIN,OMMOD,VPMOD,AMPLTD,PHASQ)
C
C CONTINUING FROM 519 OR 520
  580 IF( NPNCH .LE. 0) GO TO 590
C
C PUNCH NAM9 DATA IF NPNCH .GT. 0
      KLAST = KFIN(MOFNO)
      WRITE (7,581) (AMPLTD(I),I=1,KLAST)
  581 FORMAT ( 7H &NAM9  / 12H   AMPLTD =  /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE (7,582) (PHASQ(I),I=1,KLAST)
  582 FORMAT ( 11H    PHASQ =  /
     1  ( 6X,G15.8,1H,,G15.8,1H,,G15.8,1H,,G15.8,1H, ) )
      WRITE (7,455) MOFNO,(KST(I),I=1,MOFNO)
      WRITE(7,355) (KFIN(I),I=1,MOFNO)
      WRITE (7,357)  (OMMOD(I),I=1,KLAST)
      WRITE(7,359) (VPMOD(I),I=1,KLAST)
      WRITE (7,279)
      WRITE (6,583)
  583 FORMAT( 1H /////  41H THE FOLLOWING DATA HAS JUST BEEN PUNCHED)
      WRITE (6,581) (AMPLTD(I),I=1,KLAST)
      WRITE (6,582) (PHASQ(I),I=1,KLAST)
      WRITE (6,455) MOFNO,(KST(I),I=1,MOFNO)
      WRITE(6,355) (KFIN(I),I=1,MOFNO)
      WRITE (6,357)  (OMMOD(I),I=1,KLAST)
      WRITE(6,359) (VPMOD(I),I=1,KLAST)
  584 WRITE (6,279)
C
C CONTINUING FROM 580 OR 584
  590 GO TO 615
C
C
C WE ARRIVE HERE IF NSTART=5
  600 READ (5,NAM9)
      IF(NPRNT .LE. 0) GO TO 615
      WRITE (6,601)
  601 FORMAT(1H /////  27H NAM9 HAS JUST BEEN READ IN)
  602 WRITE (6,NAM9)
C
C CONTINUING FROM 590 OR 602
  615 READ (5,NAM10)
      WRITE (6,616)
  616 FORMAT( 1H ///// 28H NAM10 HAS JUST BEEN READ IN)
      WRITE (6,NAM10)
C
C COMPUTATION OF WAVEFORM
      CALL TMPT(TFIRST,TEND,DELTT,ROBS,MOFNO,KST,KFIN,OMMOD,VPMOD,AKI,
     1AMPLTD,PHASQ,IOPT)
C
C REPEAT FOR NEXT WAVEFORM
      GO TO 1
C
C WE ARRIVE HERE IF NSTART = 6.
C ENDPLOT TERMINATES THE CALCOMP TAPE FILE.
  CALL PLOT(0.,0.,999)
      CALL EXIT
      END
```

```
      SUBROUTINE AAAA(OMEGA,AKX,AKY,C,VX,VY,A)                          AAAA
C     AAAA (SUBROUTINE)                     7/25/68   LAST CARD IN DECK IS AAAA
C                                                                       AAAA
C                     ----ABSTRACT----                                  AAAA
C                                                                       AAAA
C TITLE - AAAA                                                          AAAA
C          THIS SUBROUTINE COMPUTES THE 2-BY-2 MATRIX A OF COEFFICIENTS AAAA
C          IN THE RESIDUAL EQUATIONS                                    AAAA
C                                                                       AAAA
C             D(PHI1)/DZ = (A11)*PHI1 + (A12)*PHI2                      AAAA
C                                                                       AAAA
C             D(PHI2)/DZ = (A21)*PHI1 + (A22)*PHI2                      AAAA
C                                                                       AAAA
C          DERIVED BY A. PIERCE, J. COMP. PHYS., VOL. 1, NO. 3, 343,-366, AAAA
C          1967.  (SEE EQN. (10) OF THE PAPER.)  THE EXPLICIT EXPRESSIONS AAAA
C          FOR THE A(I,J) ARE                                           AAAA
C                                                                       AAAA
C             A(1,1) = G*(K/BOM)**2 - GAMMA*G/(2*C**2)                  AAAA
C             A(1,2) = 1 - (C*K/BOM)**2                                 AAAA
C             A(2,1) =((G*K)/(BOM*C))**2 - (BOM/C)**2                   AAAA
C             A(2,2) = -A(1,1)                                          AAAA
C                                                                       AAAA
C          WHERE GAMMA=1.4 IS THE SPECIFIC HEAT RATIO, G=.0099 KM/SEC**2 AAAA
C          IS THE ACCELERATION OF GRAVITY, C IS THE SOUND SPEED, K IS THE AAAA
C          HORIZONTAL WAVE NUMBER AND BOM IS THE DOPPLER SHIFTED ANGULAR AAAA
C          FREQUENCY                                                    AAAA
C                                                                       AAAA
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)           AAAA
C AUTHOR      - A.D.PIERCE, M.I.T., JULY,1968                           AAAA
C                                                                       AAAA
C                     ----CALLING SEQUENCE----                          AAAA
C                                                                       AAAA
C SEE SUBROUTINES ELINT, MMMM, NAMPDE, NHOFN                            AAAA
C     DIMENSION A(2,2)                                                  AAAA
C     CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)                                AAAA
C                                                                       AAAA
C NO EXTERNAL SUBROUTINES ARE REQUIRED                                  AAAA
C                                                                       AAAA
C                     ----ARGUMENT LIST----                             AAAA
C                                                                       AAAA
C     OMEGA     R*4      NO      INP                                    AAAA
C     AKX       R*4      NO      INP                                    AAAA
C     AKY       R*4      NO      INP                                    AAAA
C     C         R*4      NO      INP                                    AAAA
C     VX        R*4      NO      INP                                    AAAA
C     VY        R*4      NO      INP                                    AAAA
C     A         R*4    2-BY-2 OUT                                       AAAA
C                                                                       AAAA
C NO COMMON STORAGE IS USED                                             AAAA
C                                                                       AAAA
C                     ----INPUTS----                                    AAAA
C                                                                       AAAA
C     OMEGA          =ANGULAR FREQUENCY IN RAD/SEC                      AAAA
C     AKX            =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM AAAA
C     AKY            =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM AAAA
C     C              =SOUND SPEED IN KM/SEC                             AAAA
C     VX             =X COMPONENT OF WIND VELOCITY IN KM/SEC            AAAA
C     VY             =Y COMPONENT OF WIND VELOCITY IN KM/SEC            AAAA
C                                                                       AAAA
C                     ----OUTPUTS----                                   AAAA
C                                                                       AAAA
C     A(I,J)         =(I,J)-TH ELEMENT OF MATRIX A OF COEFFICIESTS IN THE AAAA
C                     RESIDUAL EQUATIONS AS DEFINED IN THE ABSTRACT.    AAAA
C                                                                       AAAA
```

```
C                    ----PROGRAM FOLLOWS BELOW
C
C
      DIMENSION A(2,2)
      BOMSQ=(OMEGA-AKX*VX-AKY*VY)**2
      CSQ=C*C
      T=(AKX**2+AKY**2)/BOMSQ
      A(1,1)=.0098*T-.00686/CSQ
C GAMMA*G/2 IS .00686
      A(1,2)=1.0-CSQ*T
      A(2,1)=((96.04E-6)*T-BOMSQ)/CSQ
C G**2 IS 96.04E-6 KM**2/SEC**4
      A(2,2)=-A(1,1)
      RETURN
      END.
```

```
      SUBROUTINE AKI(OM1,OM2,A1,A2,CTRIG1,STRIG1,CTRIG2,
     1 STRIG2,DELPH,AKIINT)
C      AKI (SUBROUTINE)                   8/15/68   LAST CARD IN DECK IS
C
C
C                ----ABSTRACT----
C
C TITLE - AKI
C          EVALUATION OF INTEGRAL OF A(OMEGA)*COS(PHI(OMEGA)) FROM OM1 TO
C          OM2
C
C                A(OMEGA) AND PHI(OMEGA) ARE ASSUMED TO BE LINEAR BETWEEN
C                OM1 AND OM2, FOLLOWING THE METHOD OF AKI ( J. GEOPHYS.
C                 RES., VOL. 65 (1960), PP. 729-740 ).  THE INTEGRAL IS
C                READILY EVALUATED AS
C
C                  (PHI")**(-1) * (AI + A"*(OM2-OM1)) * SIN(PHII+X)
C
C                    + PHI"**(-2) * A" * COS(PHII + X)
C
C                    - PHI"**(-1) * (AI - A" * (OM2 - OM1)) * SIN(PHII-X
C
C                    - PHI"**(-2) * A" * COS(PHI - X)
C
C                WHERE
C
C                    AI = AVERAGE VALUE OF A IN INTERVAL
C                    PHII = AVERAGE VALUE OF PHI IN INTERVAL
C                    A" = D(A) / D(OMEGA)
C                    PHI" = D(PHI) / D(OMEGA)
C                    X = PHI" * (OM2 - OM1) / 2
C
C                A SOMEWHAT MORE CONVENIENT FORMULA OBTAINABLE BY TRIGONO
C                METRIC IDENTITIES IS
C
C                  AKIINT = 2 * PHI"**(-1) * AI * SIN(X) * COS(PHII)
C
C                           + 2 * PHI"**(-2) * A" * (X * COS(X) - SIN(X))
C
C                             * SIN(PHII)
C
C                WHENEVER X IS SMALL, SIN(X)/X AND COS(X) ARE EVALUATED B
C                USING THEIR POWER SERIES REPRESENTATIONS.
C
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., AUGUST,1968
C
C
C                ----USAGE----
C
C      NO SUBROUTINES ARE CALLED
C
C FORTRAN USAGE
C
C      CALL AKI(OM1,OM2,A1,A2,CTRIG1,STRIG1,CTRIG2,STRIG2,
C     1  DELPH,AKIINT)
C
C INPUTS
C
C     OM1       LOWER LIMIT OF INTEGRATION OVER ANGULAR FREQUENCY
C       R*4     (RADIANS)
C
C     OM2       UPPER LIMIT OF INTEGRATION (RADIANS)
```

```
C        R*4
C
C     A1           VALUE OF A AT OMEGA = OM1
C        R*4
C
C     A2           VALUE OF A AT OMEGA = OM2
C        R*4
C
C     CTRIG1       COS(PHI) WHERE OMEGA = OM1
C        R*4
C
C     STRIG1       SIN(PHI) WHERE OMEGA = OM1
C        R*4
C
C     DELPH        CHANGE IN PHI OVER THE INTERVAL ( PHI(OM2) - PHI(OM1) )
C        R*4       (RADIANS)
C
C OUTPUTS
C
C     CTRIG2       COS(PHI) WHERE OMEGA = OM2
C        R*4
C
C     STRIG2       SIN(PHI) WHERE OMEGA = OM2
C        R*4
C
C     AKIINT       VALUE OF INTEGRAL DEFINED IN ABSTRACT IN UNITS OF A*OMEG
C        R*4
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
      DELOM=OM2-OM1
      DELAA=A2-A1
C
      AI=(A2+A1)/2.0
      X=DELPH/2.0
      CTRX=COS(X)
      STRX=SIN(X)
      CTRIGI=CTRIG1*CTRX-STRIG1*STRX
      STRIGI=STRIG1*CTRX+CTRIG1*STRX
      CTRIG2=CTRIGI*CTRX-STRIGI*STRX
      STRIG2=STRIGI*CTRX+CTRIGI*STRX
      IF(ABS(X)-1.0E-2) 20,20,10
   10 S1=STRX/X
      S2=(S1-CTRX)/X**2
      GO TO 30
   20 S1=1.0-(1.0/6.0)*X**2+(1.0/120.0)*X**4
      S2=(1.0/3.0)-(1.0/30.0)*X**2+(1.0/840.0)*X**4
   30 AKIINT=(AI*S1*CTRIGI-DELAA*DELPH*0.25*S2*STRIGI)*DELOM
      RETURN
      END
```

```
      SUBROUTINE ALLMOD(NROW,NCOL,MAXMOD,MOFNO,OM,VP,KST,KFIN,OMMOD,     ALLMOD
     1 VPMOD,INMODE,THETK,KHOP)                                          ALLMOD
C     ALLMOD (SUBROUTINE)                  6/25/68   LAST CARD IN DECK IS ALLMOD
C                                                                         ALLMOD
C                                                                         ALLMOD
C TITLE - ALLMOD                                                          ALLMOD
C           PROGRAM TO TABULATE DISPERSION CURVES OF UP TO MAXMOD GUIDED  ALLMOD
C           MODES.  ONLY PORTIONS OF CURVES WITH OMEGA BETWEEN OM(1) AND  ALLMOD
C           OM(NCOL) AND WITH PHASE VELOCITY BETWEEN VP(NCOL) AND VP(1)   ALLMOD
C           ARE TABULATED.  THE ANGULAR DEVIATION OF GROUP VELOCITY DIREC- ALLMOD
C           TION FROM PHASE VELOCITY DIRECTION THETK IS NEGLECTED.        ALLMOD
C           SUCCESSIVE MODES NUMBERED FROM 1 TO MOFNO ARE EACH TABULATED B ALLMOD
C           CALLING SUBROUTINE MODETR.  STARTING POINTS FOR EACH MODE ARE ALLMOD
C           FOUND BY CALLING SUBROUTINE NXMODE.  THE NORMAL MODE DISPERSIO ALLMOD
C           FUNCTION (NMDF) SHOULD BE NEARLY ZERO FOR EVERY TABULATED POIN ALLMOD
C           ON EACH DISPERSION CURVE.  THE COMPUTATIONAL METHOD IS BASED  ALLMOD
C           ON THE PREVIOUSLY COMPUTED VALUES OF THE NMDF SIGN            ALLMOD
C           INMODE((J-1)*NROW+I) AT POINTS (I,J) IN A RECTANGULAR ARRAY OF ALLMOD
C           NROW ROWS AND NCOL COLUMNS.  DIFFERENT COLUMNS (J) CORRESPOND ALLMOD
C           TO DIFFERENT ANGULAR FREQUENCIES OM(J) WHILE DIFFERENT ROWS (I ALLMOD
C           CORRESPOND TO DIFFERENT PHASE VELOCITIES VP(I).  IT IS ASSUMED ALLMOD
C           THAT VP(1) .GT. VP(2) .GT. VP(2), ETC.  DISPERSION CURVES     ALLMOD
C           OF VARIOUS MODES APPEAR ON THIS ARRAY AS LINES OF DEMARCATION ALLMOD
C           BETWEEN ADJACENT REGIONS WITH OPPOSITE INMODES.  IT IS ASSUMED ALLMOD
C           THAT DISPERSION CURVES SLOPE DOWNWARDS.  MODES ARE NUMBERED   ALLMOD
C           STARTING FROM LOWER LEFT OF INMODE ARRAY.                     ALLMOD
C                                                                         ALLMOD
C PROGRAM NOTES                                                           ALLMOD
C                                                                         ALLMOD
C                 THE ARRAYS OMMOD AND VPMOD ARE USED TO STORE DISPERSION ALLMOD
C                 CURVES FOR ALL THE MODES TO CONSERVE STORAGE.  FOR THE  ALLMOD
C                 NMODE-TH MODE, VPMOD(KST(NMODE)+K-1) IS THE PHASE VELOCI ALLMOD
C                 CORRESPONDING TO ANGULAR FREQUENCY OF OMMOD(KST(NMODE)+ ALLMOD
C                 K-1).  THE PAIR OF VALUES CORRESPONDS TO THE K-TH TABULA ALLMOD
C                 POINT FOR THE MODE.  THE LAST TABULATED POINT FOR THE   ALLMOD
C                 NMODE-TH MODE IS LABELED BY THE PAIR VPMOD(KFIN(NMODE)), ALLMOD
C                 OMMOD(KFIN(NMODE)).  THUS OMMOD(K), VPMOD(K) FOR        ALLMOD
C                 K .GE. KST(NMODE) AND K .LT. KFIN(NMODE) DESCRIBE THE   ALLMOD
C                 NMODE-TH MODE'S DISPERSION CURVE.                       ALLMOD
C                                                                         ALLMOD
C                 THE FLAG KHOP IS NORMALLY RETURNED AS 1.  HOWEVER, IF   ALLMOD
C                 NO DISPERSION CURVES ARE TABULATED, KHOP IS RETURNED AS ALLMOD
C                 -1.                                                     ALLMOD
C                                                                         ALLMOD
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)              ALLMOD
C AUTHOR     - A.D.PIERCE, M.I.T., JUNE,1968                              ALLMOD
C                                                                         ALLMOD
C                   ----CALLING SEQUENCE----                              ALLMOD
C                                                                         ALLMOD
C SEE MAIN PROGRAM                                                        ALLMOD
C     DIMENSION OM(100),VP(100),KST(10),KFIN(10),OMMOD(1000),VPMOD(1000 ALLMOD
C     DIMENSION INMODE(10000)                                             ALLMOD
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                         ALLMOD
C THE SUBROUTINE USES VARIABLE DIMENSIONING.  THE ASSIGNMENTS ABOVE ARE ALLMOD
C THOSE GIVEN BY MAIN PROGRAM                                             ALLMOD
C     COMMON IMAX,CI,VXI,VYI,HI                                           ALLMOD
C ATMOSPHERIC VARIABLES MUST BE IN COMMON BEFORE ALLMOD IS CALLED.        ALLMOD
C     CALL ALLMOD(NROW,NCOL,MAXMOD,MOFNO,OM,VP,KST,KFIN,OMMOD,VPMOD,      ALLMOD
C    1 INMODE,THETK,KHOP                                                  ALLMOD
C     IF(KHOP .NE. 1) GO SOMEWHERE                                        ALLMOD
C                                                                         ALLMOD
C                 ----EXTERNAL SUBROUTINES REQUIRED----                   ALLMOD
C                                                                         ALLMOD
C     NXMODE,MODETR,NXTPNT,RTMI,FNMOD1,FNMOD2,NMDFN,AAAA,RRRR,MMMM,CAI,   ALLMOD
C                                                                         ALLMOD
C         NXMODE AND MODETR ARE EXPLICITLY CALLED.  THE REST ARE          ALLMOD
C         IMPLICITLY CALLED BY CALLING MODETR.  FOR FURTHER INFORMATION ALLMOD
C         ON IBM SCIENTIFIC SUBROUTINE PACKAGE ROUTINE RTMI, SEE DOCU-    ALLMOD
C         MENTATION OF MODETR.                                            ALLMOD
C                                                                         ALLMOD
C                 ----ARGUMENT LIST----                                   ALLMOD
C                                                                         ALLMOD
C     NROW         I*4      NO      INP                                   ALLMOD
C     NCOL         I*4      NO      INP                                   ALLMOD
```

```
C      MAXMOD      I*4      NO      INP                                ALLMOD
C      MDFND       I*4      NO      OUT                                ALLMOD
C      OM          R*4      VAR     INP                                ALLMOD
C      VP          R*4      VAR     INP                                ALLMOD
C      KST         I*4      VAR     OUT                                ALLMOD
C      KFIN        I*4      VAR     OUT                                ALLMOD
C      OMMOD       R*4      VAR     OUT                                ALLMOD
C      VPMOD       R*4      VAR     OUT                                ALLMOD
C      INMODE      I*4      VAR     INP                                ALLMOD
C      THETK       R*4      NO      INP                                ALLMOD
C      KWOP        I*4      NO      OUT                                ALLMOD
C                                                                      ALLMOD
C COMMON STORAGE USED                                                  ALLMOD
C      COMMON IMAX,CI,VXI,VYI,HI,OMEGAC,VPHSEC,THETKP                  ALLMOD
C                                                                      ALLMOD
C      IMAX        I*4      NO      INP                                ALLMOD
C      CI          R*4      100.    INP                                ALLMOD
C      VXI         R*4      100     INP                                ALLMOD
C      VYI         R*4      100     INP                                ALLMOD
C      HI          R*4      100     INP                                ALLMOD
C      OMEGAC      R*4      NO      OUT (USED INTERNALLY)              ALLMOD
C      VPHSEC      R*4      NO      OUT (USED INTERNALLY)              ALLMOD
C      THETKP      R*4      NO      OUT (USED INTERNALLY)              ALLMOD
C                                                                      ALLMOD
C                           ----INPUTS----                             ALLMOD
C                                                                      ALLMOD
C      NROW        =NUMBER OF ROWS IN INMODE ARRAY.  MAXIMUM INDEX OF  ALLMOD
C                   VP(N).                                             ALLMOD
C      NCOL        =NUMBER OF COLUMNS IN INMODE ARRAY.  MAXIMUM INDEX OF ALLMOD
C                   OM(N).                                             ALLMOD
C      MAXMOD      =MAXIMUM NUMBER OF MODES TO BE TABULATED            ALLMOD
C      OM(N)       =ANGULAR FREQUENCY OF N-TH COLUMN IN INMODE ARRAY   ALLMOD
C      VP(N)       =PHASE VELOCITY OF N-TH ROW IN INMODE ARRAY         ALLMOD
C      INMODE      =1,-1, OR 5 DEPENDING ON WHETHER SIGN OF NORMAL MODE ALLMOD
C                   DISPERSION FUNCTION IS + OR -, 5 IF NMDF DOESNT EXIS ALLMOD
C                   THE (J-1)*NROW+I-TH ELEMENT CORRESPONDS TO NMDF WHEN ALLMOD
C                   OMEGA=OM(J), PHASE VELOCITY=VP(I).                 ALLMOD
C      THETK       =PHASE VELOCITY DIRECTION IN RADIANS RECKONED COUNTER ALLMOD
C                   CLOCKWISE WITH RESPECT TO X AXIS.                  ALLMOD
C      IMAX        =NUMBER OF ATMOSPHERIC LAYERS OF FINITE THICKNESS   ALLMOD
C      CI(I)       =SOUND SPEED IN I-TH LAYER                          ALLMOD
C      VXI(I)      =X COMPONENT OF WIND VELOCITY IN I-TH LAYER         ALLMOD
C      VYI(I)      =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER         ALLMOD
C      HI(I)       =THICKNESS OF I-TH LAYER                            ALLMOD
C                                                                      ALLMOD
C                           ----OUTPUTS----                            ALLMOD
C                                                                      ALLMOD
C      MDFND       =NUMBER OF MODES FOUND                              ALLMOD
C      KST(N)      =INDEX OF FIRST TABULATED POINT IN N-TH MODE        ALLMOD
C      KFIN(N)     =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN    ALLMOD
C                   GENERAL, KFIN(N)=KST(N+1)-1.                       ALLMOD
C      OMMOD(N)    =ARRAY STORING ANGULAR FREQUENCY ORDINATE OF POINTS ALLMOD
C                   ON DISPERSION CURVES.  THE NMODE MODE IS STORED FOR ALLMOD
C                   N BETWEEN KST(NMODE) AND KFIN(NMODE).              ALLMOD
C      VPMOD(N)    =ARRAY STORING PHASE VELOCITY ORDINATE OF POINTS ON ALLMOD
C                   DISPERSION CURVES.  THE NMODE-TH MODE IS STORED FOR ALLMOD
C                   N BETWEEN KST(NMODE) AND KFIN(NMODE).              ALLMOD
C      KWOP        =-1 IF NO MODES ARE TABULATED.  OTHERWISE IT IS 1.  ALLMOD
C      OMEGAC      =INTERNALLY USED FREQUENCY TRANSMITTED AMONG SUBROUTI ALLMOD
C                   THROUGH COMMON                                     ALLMOD
C      VPHSEC      =INTERNALLY USED PHASE VELOCITY TRANSMITTED AMONG   ALLMOD
C                   SUBROUTINES THROUGH COMMON                         ALLMOD
C      THETKP      =SAME AS THETK                                      ALLMOD
C                                                                      ALLMOD
C                           ----EXAMPLE----                            ALLMOD
C                                                                      ALLMOD
C      SUPPOSE THE TABLE OF INMODE VALUES IS AS SHOWN BELOW WITH       ALLMOD
C                                                                      ALLMOD
C          +++--++--+       NROW=6, NCOL=10                            ALLMOD
C          +++---++--                                                  ALLMOD
```

```
C          ++++------++     IF MAXMOD=10, YOU SHOULD FIND MOFND=6.            ALLMOD
C          5---+++---                                                          ALLMOD
C          5++-++++++        KST(1)=1    KFIN(1)=4      OMMOD(1-36)  SHOULD B ALLMOD
C          5++-++++++        KST(2)=5    KFIN(2)=10     VPMOD(1-36)  TABULATE ALLMOD
C                            KST(3)=11   KFIN(3)=21                           ALLMOD
C                            KST(4)=22   KFIN(4)=29                           ALLMOD
C                            KST(5)=30   KFIN(5)=34                           ALLMOD
C                            KST(6)=35   KFIN(6)=36                           ALLMOD
C                                                                             ALLMOD
C                                                                             ALLMOD
C                   ----PROGRAM FOLLOWS BELOW----                             ALLMOD
C                                                                             ALLMOD
C                                                                             ALLMOD
C                                                                             ALLMOD
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)                             ALLMOD
      DIMENSION OM(1),VP(1),KST(1),KFIN(1),OMMOD(1),VPMOD(1),INMODE(1)        ALLMOD
      COMMON IMAX,CI,VXI,VYI,HI,OMEGAC,VPHSEC,THETKP                          ALLMOD
C                                                                             ALLMOD
C STORE THETK IN COMMON                                                       ALLMOD
      THETKP=THETK                                                            ALLMOD
C                                                                             ALLMOD
C AT THIS POINT, WE HAVEN-T FOUND ANY MODES                                   ALLMOD
      MOFND=0                                                                 ALLMOD
C                                                                             ALLMOD
C WE START SEARCH FOR FIRST MODE IN LOWER LEFT CORNER OF INMODE ARRAY.        ALLMOD
C WE SEEK A POINT WITH INMODE .NE. 5 WHERE THE NMDF EXISTS.                   ALLMOD
      NMODE=1                                                                 ALLMOD
      KST(NMODE)=1                                                            ALLMOD
      IST=NROW                                                                ALLMOD
C                                                                             ALLMOD
C THE SEARCH GOES TO THE RIGHT.  IF WE DON-T FIND A POINT IN THE BOTTOM       ALLMOD
C ROW, WE TRY THE (NROW-1)-TH ROW, ETC.  AT STATEMENT 2 WE ARE STARTING       ALLMOD
C AT THE LEFT OF A GIVEN ROW.                                                 ALLMOD
    2 JST=1                                                                   ALLMOD
    3 J50=(JST-1)*NROW+IST                                                    ALLMOD
      I0=INMODE(J50)                                                          ALLMOD
      IF(I0 .NE. 5) GO TO 10                                                  ALLMOD
C                                                                             ALLMOD
C IF JST IS NOT NCOL WE GO TO THE RIGHT.                                      ALLMOD
      IF(JST .EQ. NCOL) GO TO 5                                               ALLMOD
      JST=JST+1                                                               ALLMOD
      GO TO 3                                                                 ALLMOD
C                                                                             ALLMOD
C AT THIS POINT WE HAVE EXHAUSTED AN ENTIRE ROW.  WE GO TO THE NEXT           ALLMOD
C HIGHER ROW PROVIDED IST .NE. 1.  IF IST IS 1, THE ENTIRE SET OF             ALLMOD
C INMODES ARE 5.                                                              ALLMOD
    5 IF(IST .EQ. 1) GO TO 7                                                  ALLMOD
      IST=IST-1                                                               ALLMOD
      GO TO 2                                                                 ALLMOD
C                                                                             ALLMOD
    7 WRITE (6,8)                                                             ALLMOD
    8 FORMAT(1H0,51HTHE NORMAL MODE DISPERSION FUNCTION DOES NOT EXIST        ALLMOD
     1 26HFOR ANY POINT IN THE ARRAY /  1H ,22HALLMOD RETURNS KWOP=-1)        ALLMOD
    9 KWOP=-1                                                                 ALLMOD
      RETURN                                                                  ALLMOD
C                                                                             ALLMOD
C STATEMENT 10 IS START OF LOOP.  EACH PASSAGE THROUGH LOOP CORRESPONDS       ALLMOD
C TO A GIVEN MODE.                                                            ALLMOD
   10 CALL NXMODE(IST,JST,NCOL,NROW,INMODE,IFND,JFND,KEX)                     ALLMOD
C                                                                             ALLMOD
C IF YOU CANNOT FIND THE FIRST MODE YOU ARE IN TROUBLE                        ALLMOD
      IF(NMODE .NE. 1) GO TO 15                                               ALLMOD
      IF(KEX .EQ. 1) GO TO 15                                                 ALLMOD
      WRITE (6,11)                                                            ALLMOD
```

```
   11 FORMAT(1H0,36HNXMODE COULD NOT FIND THE FIRST MODE/ 1H ,
     122HALLMOD RETURNS KWOP=-1)
      GO TO 9
C
C IF THE MODE SOUGHT IS NOT THE FIRST AND YOU CANNOT FIND IT, THEN THE
C RETURN IS CONSIDERED SATISFACTORY.
   15 IF(KEX .EQ. -1) GO TO 50
C
C WE NOW TABULATE THE NMODE-TH MODE
      CALL MODETR(IFND,JFND,NMODE,KST,KFIN,OMMOD,VPMOD,NROW,NCOL,INMODE
     1 OM,VP,KRUD)
C
C IT IS DOUBFUL THAT KRUD COULD BE -1.  HOWEVER, IF IT DID HAPPEN, WE
C WOULD LIKE TO KNOW THAT IT DID.
      IF(KRUD .EQ. 1) GO TO 30
      WRITE (6,21) NMODE,IFND,JFND
   21 FORMAT(1H0,23HMODETR RETURNS KRUD=-1.,2X,25HCURRENT VALUE OF NMOD
     1 IS, I4, 3H.   , 5HIFND=, I4,3H,   , 5HJFND=, I4/  1H ,27HSEE DOCUM
     2NTATION OF ALLMOD)
C
C WE KEEP NMODE THE SAME AND TRANSFER CONTROL TO STATEMENT 35
      GO TO 35
   30 MOFND=MOFND+1
C
C THIS IS THE CURRENT NUMBER OF MODES FOUND.
C WE NOW CHECK IF THIS IS MAXMOD.  IF IT IS, THE RETURN IS WITH KWOP=1.
      IF(MOFND .EQ. MAXMOD) GO TO 50
      NMODE=NMODE+1
      KST(NMODE)=KFIN(NMODE-1)+1
C
C WE SEEK NEW IST AND JST BEFORE CALLING NXMODE.
   35 J52=(JFND-1)*NROW+IFND
      I0=INMODE(J52)
      IF(IFND .EQ. 1) GO TO 40
C
C WE CHECK INMODE OF POINT ABOVE
      J3=(JFND-1)*NROW+IFND-1
      IUP=INMODE(J3)
C
C IF THIS IS -I0, THE POINT ABOVE IS THE ONE DESIRED
      IF(IUP .NE. -I0) GO TO 40
      IST=IFND-1
      JST=JFND
      GO TO 10
C
C WE CHECK INMODE OF POINT TO RIGHT.  THERE IS NO PLACE TO GO IF JFND=
C NCOL.  THIS IS INTERPRETED AS SUCCESS PROVIDING MOFND .NE. 0.
   40 IF(JFND .NE. NCOL) GO TO 43
      GO TO 50
C
C IRT IS INMODE OF POINT TO RIGHT
   43 J4=(JFND)*NROW+IFND
      IRT=INMODE(J4)
      IF(IRT .NE. -I0) GO TO 50
      IST=IFND
      JST=JFND+1
      GO TO 10
C
C THE SEARCH HAS TERMINATED.  IF MOFND=0, WE HAVE BEEN UNSUCCESSFUL.
   50 IF(MOFND .EQ. 0) GO TO 9
      KWOP=1
      RETURN
      END
```

```
      SUBROUTINE AMBNT(Z,PRESUR,I)                                      AMBNT
C     AMBNT (SUBROUTINE)                 7/27/68   LAST CARD IN DECK IS AMBNT
C                                                                       AMBNT
C                   ----ABSTRACT----                                    AMBNT
C                                                                       AMBNT
C TITLE - AMBNT                                                         AMBNT
C          THIS SUBROUTINE COMPUTES THE AMBIENT PRESSURE IN DYNES/CM**2 AMBNT
C          AT A GIVEN ALTITUDE Z KM BY USE OF THE EQUATION              AMBNT
C                                                                       AMBNT
C            PRESUR = (1.E6)*EXP(-INTEGRAL FROM 0 TO Z OF GAMMA*G/C**2) AMBNT
C                                                                       AMBNT
C          WHERE 1.E6 DYNES/CM**2 IS THE AMBIENT PRESSURE AT THE GROUND. AMBNT
C          GAMMA=1.4 IS THE SPECIFIC HEAT RATIO FOR AIR, G=.0098 KM/SEC** AMBNT
C          IS THE ACCELERATION OF GRAVITY, AND C IS THE ALTITUDE DEPENDEN AMBNT
C          SOUND SPEED IN KM/SEC.  THE ABOVE EQUATION FOLLOWS FROM THE  AMBNT
C          HYDROSTATIC EQUATION D(P0)/DZ = -G*RHO0 AND THE IDEAL GAS LAW AMBNT
C          C**2 = GAMMA*P0/RHO0.                                        AMBNT
C                                                                       AMBNT
C          THE SOUND SPEED PROFILE IS THAT OF A MULTILAYER ATMOSPHERE AND AMBNT
C          IS PRESUMED TO BE STORED IN COMMON BEFORE EXECUTION.  THE    AMBNT
C          PROGRAM ALSO RETURNS THE INDEX I OF THE LAYER IN WHICH Z LIES. AMBNT
C                                                                       AMBNT
C PROGRAM NOTES                                                         AMBNT
C                                                                       AMBNT
C                IN THE EVENT THAT THE INPUT VALUE OF Z SHOULD BE NEGATIV AMBNT
C                THE FIRST LAYER IS ASSUMED TO HOLD FOR Z .LT. 0 WITH THE AMBNT
C                AMBIENT PRESSURE STILL EQUAL TO 1.E6 AT Z=0.  THE PROGRA AMBNT
C                RETURNS PRESUR .GT. 1.E6 AND I=1.                      AMBNT
C                                                                       AMBNT
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)            AMBNT
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968                            AMBNT
C                                                                       AMBNT
C                   ----CALLING SEQUENCE----                            AMBNT
C                                                                       AMBNT
C SEE SUBROUTINE PAMPOE                                                 AMBNT
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                       AMBNT
C     COMMON IMAX,CI,VXI,VYI,HI    (THESE MUST BE STORED IN COMMON)     AMBNT
C     CALL AMBNT(Z,PRESUR,I)                                            AMBNT
C                                                                       AMBNT
C                   ----EXTERNAL SUBROUTINES REQUIRED----               AMBNT
C                                                                       AMBNT
C     NO EXTERNAL SUBROUTINES ARE REQUIRED.                             AMBNT
C                                                                       AMBNT
C                   ----ARGUMENT LIST----                               AMBNT
C                                                                       AMBNT
C     Z            R*4     NO     INP                                   AMBNT
C     PRESUR       R*4     NO     OUT                                   AMBNT
C     I            I*4     NO     OUT                                   AMBNT
C                                                                       AMBNT
C COMMON STORAGE USED                                                   AMBNT
C     COMMON IMAX,CI,VXI,VYI,HI                                         AMBNT
C                                                                       AMBNT
C     IMAX         I*4     NO     INP                                   AMBNT
C     CI           R*4     100    INP                                   AMBNT
C     VXI          R*4     100    INP   (NOT USED BY THIS SUBROUTINE)   AMBNT
C     VYI          R*4     100    INP   (NOT USED BY THIS SUBROUTINE)   AMBNT
C     HI           R*4     100    INP                                   AMBNT
C                                                                       AMBNT
C                   ----INPUTS----                                      AMBNT
C                                                                       AMBNT
C     Z           =HEIGHT IN KM                                         AMBNT
```

```
C      IMAX        =NUMBER OF ATMOSPHERIC LAYERS WITH FINITE THICKNESS
C      CI(I)       =SOUND SPEED (KM/SEC) IN I-TH LAYER
C      VXI(I)      =X COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C      VYI(I)      =Y COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C      HI(I)       =THICKNESS IN KM OF I-TH LAYER
C
C                   ----OUTPUTS----
C
C      PRESUR      =AMBIENT PRESSURE IN DYNES/CM**2 AT ALTITUDE Z
C      I           =INDEX OF LAYER IN WHICH Z LIES
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C DIMENSION AND COMMON STATEMENTS
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      COMMON IMAX,CI,VXI,VYI,HI
C
C THE FINAL VALUE OF ENPON WILL BE THE INTEGRAL FROM 0 TO Z OF
C -GAMMA*G/C**2.  THE RUNNING VALUE WILL BE THE SUBTOTAL.
      ENPON=0.0
C
C THE RUNNING VALUE OF I WILL BE THE LAYER BEING CONSIDERED
      I=1
C Z LIES IN LAYER 1 IF IMAX=0.
      ZT=0.0
      IF(IMAX .EQ. 0) GO TO 30
C
C TOP OF FIRST LAYER
      ZT=HI(1)
C
C THE START OF A LOOP.  THE CURRENT ZT DENOTES THE TOP OF THE I-TH LAYE
   10 IF( Z .GT. ZT ) GO TO 20
C
C Z LIES IN I-TH LAYER
C ZT-HI(I) IS HEIGHT OF BOTTOM OF I-TH LAYER
C Z-ZT+HI(I) IS DISTANCE OF Z ABOVE BOTTOM OF I-TH LAYER
      ENPON=ENPON-1.4*(.0098/CI(I)**2)*(Z-ZT+HI(I))
   12 GO TO 40
C
C Z LIES ABOVE TOP OF I-TH LAYER
   20 ENPON=ENPON-1.4*(.0098/CI(I)**2)*HI(I)
C THE CURRENT ENPON IS THE INTEGRAL OF -1.4*G/C**2 UP TO THE TOP
C OF THE I-TH LAYER
      I=I+1
      IF(I .GT. IMAX) GO TO 30
      ZT=ZT+HI(I)
C ZT IS THE TOP OF THE NEW I-TH LAYER
      GO TO 10
C END OF LOOP
C
C Z LIES IN UPPER HALFSPACE
   30 ENPON=ENPON-1.4*(.0098/CI(I)**2)*(Z-ZT)
C
C CONTINUING FROM 12 OR 30
   40 PRESUR=1.E6*EXP(ENPON)
      RETURN
      END
```

```
      SUBROUTINE ATMOS(T,VKNTX,VKNTY,ZI,WANGLE,WINDY,LANGLE)              ATMOS
C     ATMOS (SUBROUTINE)                    6/19/68   LAST CARD IN DECK IS ATMOS
C                                                                          ATMOS
C                                                                          ATMOS
C                        ----ABSTRACT----                                  ATMOS
C                                                                          ATMOS
C                                                                          ATMOS
C TITLE - ATMOS                                                            ATMOS
C          TABULATION OF WIND VELOCITY COMPONENTS AND SPEED OF SOUND FOR  ATMOS
C          ALL LAYERS OF MODEL ATMOSPHERES                                 ATMOS
C                                                                          ATMOS
C                 THE MODEL ATMOSPHERE CONSISTS OF UP TO 100 ISOTHERMAL   ATMOS
C                 LAYERS (THE TOP LAYER BEING INFINITE).  EACH LAYER MAY  ATMOS
C                 HAVE A UNIQUE TEMPERATURE, THICKNESS AND WIND VELOCITY. ATMOS
C                 SUBROUTINE ATMOS CONVERTS AN INPUT DESCRIPTION OF THE   ATMOS
C                 ATMOSPHERE"S PROPERTIES INTO ONE MORE APPROPRIATE FOR TH ATMOS
C                 CALCULATIONS TO FOLLOW  (SUCH AS EVALUATION OF THE NORMA ATMOS
C                 MODE DISPERSION FUNCTION IN NMDFN, DESCRIBED ELSEWHERE I ATMOS
C                 THIS SERIES).                                            ATMOS
C                                                                          ATMOS
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             ATMOS
C                                                                          ATMOS
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                 ATMOS
C                                                                          ATMOS
C                                                                          ATMOS
C                        ----USAGE----                                     ATMOS
C                                                                          ATMOS
C     IMAX MUST BE STORED AS THE FIRST VARIABLE IN UNLABELED COMMON WHE   ATMOS
C     ATMOS IS CALLED.                                                     ATMOS
C                                                                          ATMOS
C     NO FORTRAN SUBROUTINES ARE CALLED.                                   ATMOS
C                                                                          ATMOS
C FORTRAN USAGE                                                            ATMOS
C                                                                          ATMOS
C     CALL ATMOS(T,VKNTX,VKNTY,ZI,WANGLE,WINDY,LANGLE)                     ATMOS
C                                                                          ATMOS
C INPUTS                                                                   ATMOS
C                                                                          ATMOS
C    IMAX        NUMBER OF LAYERS OF FINITE THICKNESS IN THE MODEL ATMOS- ATMOS
C      I*4       PHERE.  ( 1.LE.IMAX.LE.99 )                               ATMOS
C                                                                          ATMOS
C    T           T(I) IS TEMPERATURE OF LAYER I IN MODEL ATMOSPHERE.      ATMOS
C      R*4(D)    (DEGREES KELVIN)                                          ATMOS
C                                                                          ATMOS
C    VKNTX       VKNTX(I) IS WIND VELOCITY COMPONENT IN X-DIRECTION (WEST ATMOS
C      R*4(D)    TO EAST) FOR LAYER I.  (KNOTS)                           ATMOS
C                                                                          ATMOS
C    VKNTY       VKNTY(I) IS WIND VELOCITY COMPONENT IN Y-DIRECTION (SOUT ATMOS
C      R*4(D)    TO NORTH) FOR LAYER I.  (KNOTS)                          ATMOS
C                                                                          ATMOS
C    ZI          ZI(I) IS THE HEIGHT ABOVE THE GROUND OF THE TOP OF LAYER ATMOS
C      R*4(D)    I.  (KM)                                                  ATMOS
C                                                                          ATMOS
C    WANGLE      WANGLE(I) IS WIND VELOCITY DIRECTION FOR LAYER I, RECKON ATMOS
C      R*4(D)    COUNTER CLOCKWISE FROM THE X-AXIS.  (DEGREES)            ATMOS
C                                                                          ATMOS
C    WINDY       WINDY(I) IS MAGNITUDE OF WIND VELOCITY IN LAYER I.       ATMOS
C      R*4(D)    (KNOTS)                                                   ATMOS
C                                                                          ATMOS
C    LANGLE      SPECIFIES WHICH SORT OF WIND DATA IS INPUT.              ATMOS
C      I*4       IF LANGLE.LE.0 , VKNTX AND VKNTY ARE INPUT.              ATMOS
C                IF LANGLE.GT.0 , WANGLE AND WINDY ARE INPUT.             ATMOS
C                                                                          ATMOS
C OUTPUTS                                                                  ATMOS
C           THE OUTPUTS ARE STORED IN UNLABELED COMMON IN THE FOLLOWING   ATMOS
C           ORDER, BEGINNG IN POSITION 2.                                 ATMOS
C           CI(100),VXI(100),VYI(100),HI(100)                             ATMOS
C                                                                          ATMOS
C    CI          CI(I) IS THE SPEED OF SOUND IN LAYER I OF THE MODEL ATMO ATMOS
C      R*4(D)    PHERE.  ( KM/SEC )                                        ATMOS
```

```
C
C     VXI       VXI(I) IS WINC VELOCITY COMPONENT IN X-CIRECTION (WEST T
C       R*4(0)  EAST) FOR LAYER I.  ( KM/SEC )
C
C     VYI       VYI(I) IS WING VELOCITY COMPONENT IN Y-CIRECTION (SOUTH
C       R*4(0)  TO NORTH) FOR LAYER I.  ( KM/SEC )
C
C     HI        HI(I) IS THE THICKNESS OF LAY.  I.  ( KM )
C       R*4(0)
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      DIMENSION T(100),VKNTX(100),VKNTY(100),ZI(100)
      DIMENSION WANGLE(100),WINDY(100)
      COMMON IMAX,CI,VXI,VYI,HI
C
C JET IS TOTAL NUMBER OF LAYERS.
      JET = IMAX + 1
      IMAX = JET - 1
      IF (LANGLE .LE. 0) GO TO 20
      D3 = 3.1415927 / 180.0
C D3 IS THE NUMBER OF RADIANS IN A DEGREE
C
C IF VKNTX AND VKNTY WERE NOT INPUT, THEY ARE NOW DETERMINED FROM WINDY
C AND WANGLE.
      DO 5  I=1,JET
      VKNTX(I) = WINDY(I) * COS(D3*WANGLE(I))
    5 VKNTY(I) = WINDY(I) * SIN(D3*WANGLE(I))
   20 D1 = 1.4 * 8.3144 * 0.001 / 29.0
C D2 IS THE NUMBER OF KM/SEC PER KNOT.
      D2 = 0.0005148
C
      DO 30  I = 1,JET
C
C THE SPEED OF SOUND = ( GAMMA * P / RHO ) FOR PERFECT GAS, AND ( P/RHO
C = ( R * T )
C R IS THE (UNIVERSAL GAS CONSTANT)/(MOLECULAR WEIGHT)
      CI(I) = SQRT(D1*T(I))
C
C ( D2 * V(KNOTS) )  =  V(KM/SEC)
      VXI(I) = D2 * VKNTX(I)
   30 VYI(I) = D2 * VKNTY(I)
      IF( IMAX .EQ. 0) RETURN
      HI(1) = ZI(1)
      IF(IMAX .EQ. 1) RETURN
      DO 40  I=2,IMAX
   40 HI(I) = ZI(I) - ZI(I-1)
      RETURN
      END
```

```
      SUBROUTINE B889(X,R1,R2,R3)                                       B889
C     B889 (SUBROUTINE)                        7/25/68   LAST CARD IN DECK IS B889
C                                                                       B889
C                     ----ABSTRACT----                                  B889
C                                                                       B889
C TITLE - B889                                                          B889
C         THIS SUBROUTINE COMPUTES THREE FUNCTIONS R1,R2,R3 OF A VARIABL B889
C         X.  THESE ARE DEFINED FOR X .GE. 0 BY THE FORMULAS            B889
C                                                                       B889
C            R1= 1.0 +SINH(2Y)/(2Y)                                     B889
C                                                                       B889
C            R2= (SINH(2Y)/2Y - 1.0)/Y**2                               B889
C                                                                       B889
C            R3= (COSH(2Y)-1.0)/Y**2                                    B889
C                                                                       B889
C         WHERE Y= SQRT(X).  FORMULAS FOR NEGATIVE X MAY BE OBTAINED BY B889
C         ANALYTIC CONTINUATION.  FOR SMALL VALUES OF X, THE FUNCTIONS  B889
C         ARE COMPUTABLE BY THE POWER SERIES -                          B889
C                                                                       B889
C            R1= 2 + 4X/(3FACT) +(4X)**2/(5FACT) + (4X)**3/(7FACT) +... B889
C                                                                       B889
C            R2= 4/(3FACT) + 4*(4X)/(5FACT) + 4*(4X)**2/(7FACT) +...    B889
C                                                                       B889
C            R3= 4/(2FACT) + 4*(4X)/(4FACT) + 4*(4X)**2/(6FACT) +...    B889
C                                                                       B889
C         THE MANNER IN WHICH THESE PARTICULAR FUNCTIONS ARISE IN THE   B889
C         THEORY COMES FROM INTEGRATIONS OVER VARIOUS PRODUCTS OF CAI(X) B889
C         AND SAI(X).  IN PARTICULAR, FOR X POSITIVE,                   B889
C                                                                       B889
C            R1= (2/Y)(INTEGRAL ON Y FROM 0 TO Y OF (COSH(Y))**2)       B889
C                                                                       B889
C            R2= (2/Y**3)(INTEGRAL ON Y FROM 0 TO Y OF (SINH(Y))**2)    B889
C                                                                       B889
C            R3= (4/Y**2)(INTEGRAL ON Y FROM 0 TO Y OF SINH(Y)*COSH(Y)) B889
C                                                                       B889
C         WITH Y=SQRT(X).  THE CORRESPONDING FORMULAS FOR X NEGATIVE CAN B889
C         BE OBTAINED BY REPLACING SINH AND COSH BY SIN AND COS, RESPEC- B889
C         TIVELY, AND BY REINTERPRETING Y AS SQRT(-X).                  B889
C                                                                       B889
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            B889
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968                            B889
C                                                                       B889
C                     ----CALLING SEQUENCE----                          B889
C                                                                       B889
C SEE SUBROUTINE ELINT                                                  B889
C     X=                                                                B889
C     CALL B889(X,R1,R2,R3)                                             B889
C                                                                       B889
C                     ----EXTERNAL SUBROUTINES REQUIRED----             B889
C                                                                       B889
C     CAI, SAI                                                          B889
C                                                                       B889
C                     ----ARGUMENT LIST----                             B889
C                                                                       B889
C     X          R*4      NO     INP                                    B889
C     R1         R*4      NO     OUT                                    B889
C     R2         R*4      NO     OUT                                    B889
C     R3         R*4      NO     OUT                                    B889
C                                                                       B889
C NO COMMON STORAGE IS USED                                             B889
C                                                                       B889
C                     ----PROGRAM FOLLOWS BELOW----                     B889
C                                                                       B889
      S=SAI(4.0*X)                                                      B889
```

```
      IF(ABS(X) .GT. 1.E-2) GO TO 3
C
C COMPUTATION FOR SMALL X
      R2=2.0/3.0+(2.0/15.0)*X+(4.0/315.0)*X**2+(2.0/9.0)*X**3/315.0
      R3=2.0+2.0*X/3.0+4.0*X**2/45.0+2.0*X**3/315.0
      GO TO 4
C
C COMPUTATION FOR X NOT NEAR ZERO
    3 R2=(S-1.0)/X
      R3=(CAI(4.0*X)-1.0)/X
C
C COMPUTATION CF R1 FOR ARBITRARY X
    4 R1=1.0+S
      RETURN
      END
```

```
      FUNCTION CAI(X)                                                   CAI
C     CAI (FUNCTION)                        7/25/68   LAST CARD IN DECK IS CAI
C                                                                       CAI
C                    ----ABSTRACT----                                   CAI
C                                                                       CAI
C TITLE - CAI                                                           CAI
C         PROGRAM TO EVALUATE FUNCTION CAI(X) FOR GIVEN VARIABLE X.     CAI
C         IF X IS NEGATIVE, CAI(X)= COS(SQRT(-X)).  IF X IS POSITIVE,   CAI
C         CAI(X)= COSH(SQRT(+X)).  THE FUNCTION IS ALSO REPRESENTABLE   CAI
C         BY THE POWER SERIES                                           CAI
C                                                                       CAI
C             CAI(X)= 1 + X/(2FACT) + X**2/(4FACT) + X**3/(6FACT) + ... CAI
C                                                                       CAI
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            CAI
C                                                                       CAI
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968                            CAI
C                    ----CALLING SEQUENCE----                           CAI
C                                                                       CAI
C     CAI(ANY R*4 ARGUMENT) MAY BE USED IN ARITHMETIC EXPRESSIONS       CAI
C                                                                       CAI
C                    ----EXTERNAL SUBROUTINES REQUIRED----              CAI
C                                                                       CAI
C     NO EXTERNAL SUBROUTINES ARE REQUIRED                              CAI
C                                                                       CAI
C                    ----ARGUMENT LIST----                              CAI
C                                                                       CAI
C     X          R*4     NO     INP                                     CAI
C     CAI        R*4     NO     OUT                                     CAI
C                                                                       CAI
C NO COMMON STORAGE IS USED                                             CAI
C                                                                       CAI
C                    ----PROGRAM FOLLOWS BELOW----                      CAI
C                                                                       CAI
C                                                                       CAI
      IF(X .GE. 0.0) GO TO 11                                           CAI
C                                                                       CAI
C X IS LESS THAN 0                                                      CAI
   10 CAI=COS(SQRT(-X))                                                 CAI
      RETURN                                                            CAI
C                                                                       CAI
C X IS GREATER OR EQUAL TO 0                                            CAI
   11 E=EXP(SQRT(X))                                                    CAI
C THE HYPERBOLIC COSINE IS COMPUTED                                     CAI
      CAI=0.5*(E+1./E)                                                  CAI
      RETURN                                                            CAI
      END                                                               CAI
```

```
      SUBROUTINE DADQR(OMEGA,AKX,AKY,C,VX,VY,DADOM,DADKX,DADKY)
C     DADQR  (SUBROUTINE)  MODIFIED 7/11/74  LAST CARD IN DECK IS NO.
C
C
C                  ----ABSTRACT----
C
C TITLE - DADQR
C          THE FUNCTION OF THIS SUBROUTINE IS TO COMPUTE THE COMPONENTS
C          OF THE MATRICES DADOM,DADKX, AND DADKY WHICH REPRESENT THE
C          PARTIAL DERIVATIVES OF THE MATRIX A WHICH WOULD BE
C          COMPUTED BY SUBROUTINE AAAA.
C               DADOM IS THE PARTIAL DERIVATIVE MATRIX OF A WRT OMEGA
C               DADKX IS THE PARTIAL DERIVATIVE MATPIX OF A WRT AKX
C               DADKY IS THE PARTIAL DERIVATIVE MATRIX OF A WRT AKY
C          LIKE A, ALL ARE 2-BY-2 MATRICES.
C
C LANGUAGE - FORTRAN V (UNIVAC 1108,REFERENCE MANUAL UP-7536 REV. 1)
C AUTHORS - ALLAN D. PIERCE, CHRISTOPHER KAPPER, G.I.T., JULY, 1974
C
C                  ----CALLING SEQUENCE----
C
C SEE SUBROUTINE COMPK
C     DIMENSION D(2,2),DADOM(2,2),DADKX(2,2),DADKY(2,2)
C     CALL DADQR(OMEGA,AKX,AKY,C,VX,VY,DADOM,DADKX,DADKY)
C
C NO EXTERNAL SUBROUTINES REQUIRED
C
C                  ----ARGUMENT LIST----
C
C     OMEGA      R*4     NO       INP
C     AKX        R*4     NO       INP
C     AKY        R*4     NO       INP
C     C          R*4     NO       INP
C     VX         R*4     NO       INP
C     VY         R*4     NO       INP
C     DADOM      R*4     2-BY-2 OUT
C     DADKX      R*4     2-BY-2 OUT
C     DADKY      R*4     2-BY-2 OUT
C
C NO COMMON STORAGE USED
C
C                  ----INPUTS----
C
C     OMEGA       =ANGULAR FREQUENCY  RAD/SEC
C     AKX         =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     AKY         =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     C           =SOUND SPEED IN KM/SEC
C     VX          =X COMPONENT OF WIND VELOCITY IN KM/SEC
C     VY          =Y COMPONENT OF WIND VELOCITY IN KM/SEC
C
C                  ----OUTPUTS----
C
C     DADOM(I,J)  =(I,J)-TH ELEMENT OF DADOM MATRIX
C     DADKX(I,J)  =(I,J)-TH ELEMENT OF DADKX MATRIX
C     DADKY(I,J)  =(I,J)-TH ELEMENT OF DADKY MATRIX
C
C                  ----PROGRAM FOLLOWS BELOW----
C
C     DADOM,DADKX,DADKY ARE MATRIX DERIVATIVES OF A WITH RESPECT TO
```

```
      OMEGA,AKX,AKY, WHERE A IS AS COMPUTED BY AAAA.                    DADQR
      DIMENSION D(2,2),DADOM(2,2),DADKX(2,2),DADKY(2,2)                 DADQR
      CSQ=C*C                                                           DADQR
      D(1,1)=.0098                                                      DADQR
      D(1,2)=-CSQ                                                       DADQR
      D(2,1)=(96.04E-6)/CSQ                                             DADQR
      D(2,2)=-.0098                                                     DADQR
      BOM=OMEGA-AKX*VX-AKY*VY                                           DADQR
      BOMSQ=BOM**2                                                      DADQR
C     T IS AKSQ/BOMSQ                                                   DADQR
      DTDOM=-2.0*(AKX**2+AKY**2)/(BOMSQ*BOM)                            DADQR
      DTDKX=-DTDOM*VX+2.0*AKX/BOMSQ                                     DADQR
      DTDKY=-DTDOM*VY+2.0*AKY/BOMSQ                                     DADQR
      DO 90 I=1,2                                                       DADQR
      DO 90 J=1,2                                                       DADQR
      DADOM(I,J)=DTDOM*D(I,J)                                           DADQR
      DADKX(I,J)=DTDKX*D(I,J)                                           DADQR
   90 DADKY(I,J)=DTDKY*D(I,J)                                           DADQR
C     THE ABOVE ELEMENTS ARE CORRECT EXCEPT FOR (2,1) ELEMENTS          DADQR
      XAT=2.0*BOM/CSQ                                                   DADQR
C     XAT IS THE DERIVATIVE WITH RESPECT TO OMEGA OF BOMSQ/CSQ          DADQR
      DADOM(2,1)=DADOM(2,1)-XAT                                         DADQR
      DADKX(2,1)=DADKX(2,1)+XAT*VX                                      DADQR
      DADKY(2,1)=DADKY(2,1)+XAT*VY                                      DADQR
      RETURN                                                            DADQR
      END                                                               DADQR
```

```
      SUBROUTINE DFDOR(OMEGA,AKX,AKY,GI,RPP,A,C,VX,VY,DFDOM,DFDKX,DFDKY)DFDOR
C.    DFDOR  (SUBROUTINE)  MODIFIED 7/11/74  LAST CARD IN DECK IS NO.   DFDOR
C                                                                       DFDOR
C                                                                       DFDOR
C                        ----ABSTRACT----                               DFDOR
C                                                                       DFDOR
C TITLE - DFDOR                                                         DFDOR
C           THE NORMAL MODE DISPERSION FUNCTION COMPUTED BY SUBROUTINE  DFDOR
C           NMDFN IS CONSIDERED A FUNCTION OF OMEGA,AKX, AND AKY.       DFDOR
C           THIS SUBROUTINE COMPUTES THE PARTIAL DERIVATIVES OF FPP WITH  DFDOR
C           RESPECT TO OMEGA,AKX, AND AKY RESPECTIVELY.                 DFDOR
C                DFDOM IS THE PARTIAL DERIVATIVE OF FPP WRT OMEGA       DFDOR
C                DFDKX IS THE PARTIAL DERIVATIVE OF FPP WRT AKX         DFDOR
C                DFDKY IS THE PARTIAL DERIVATIVE OF FPP WRT AKY         DFDOR
C                                                                       DFDOR
C LANGUAGE - FORTRAN V (UNIVAC 1108,REFERENCE MANUAL UP-7536 REV. 1)    DFDOR
C                                                                       DFDOR
C AUTHORS - A.D. PIERCE, CHRISTOPHER KAPPER, G.I.T., JULY, 1974         DFDOR
C                                                                       DFDOR
C                        ----CALLING SEQUENCE----                       DFDOR
C                                                                       DFDOR
C SEE SUBROUTINE COMPK                                                  DFDOR
C     DIMENSION RPP(2,2),DADOM(2,2),DADKX(2,2),DADKY(2,2)               DFDOR
C     DIMENSION DRDOM(2,2),DRDKX(2,2),DRDKY(2,2),A(2,2)                 DFDOR
C     CALL DFDOR(OMEGA,AKX,AKY,GI,RPP,A,C,VX,VY,DFDOM,DFDKX,DFDKY)      DFDOR
C                                                                       DFDOR
C                        ----EXTERNAL SUBROUTINES REQUIRED----          DFDOR
C                                                                       DFDOR
C     DADOR,DRDOR, (DRDOR CALLS DMDOR)                                  DFDOR
C                                                                       DFDOR
C                        ----ARGUMENT LIST----                          DFDOR
C                                                                       DFDOR
C     OMEGA        R*4      NO      INP                                 DFDOR
C     AKX          R*4      NO      INP                                 DFDOR
C     AKY          R*4      NO      INP                                 DFDOR
C     GI           R*4      NO      INP                                 DFDOR
C     C            R*4      NO      INP                                 DFDOR
C     VX           R*4      NO      INP                                 DFDOR
C     VY           R*4      NO      INP                                 DFDOR
C     A            R*4      2-BY-2  INP                                 DFDOR
C     RPP          R*4      2-BY-2  INP                                 DFDOR
C     DFDOM        R*4      NO      OUT                                 DFDOR
C     DFDKX        R*4      NO      OUT                                 DFDOR
C     DFDKY        R*4      NO      OUT                                 DFDOR
C                                                                       DFDOR
C NO COMMON STORAGE USED                                                DFDOR
C                                                                       DFDOR
C                        ----INPUTS----                                 DFDOR
C                                                                       DFDOR
C     OMEGA        =ANGULAR FREQUENCY   RAD/SEC                         DFDOR
C     AKX          =X COMPONENET OF HORIZONTAL WAVE NUMBER VECTOR IN 1/K DFDOR
C     AKY          =Y COMPONENET OF HORIZONTAL WAVE NUMBER VECTOR IN 1/K DFDOR
C     GI           =PARAMETER FOR DETERMINING GU IN UPPER HALFSPACE     DFDOR
C     C            =SOUND SPEED IN KM/SEC                               DFDOR
C     VX           =X COMPONENT OF WIND VELOCITY IN KM/SEC              DFDOR
C     VY           =Y COMPONENT OF WIND VELOCITY IN KM/SEC              DFDOR
C     RPP(I,J)     =(I,J)-TH ELEMENT OF MATRIX RPP CONNECTING           DFDOR
C                   SOLUTIONS OF THE RESIDUAL EQUATIONS AT THE BOTTOM OF DFDOR
C                   THE UPPER HALFSPACE TO SOLUTIONS AT THE GROUND.     DFDOR
C     A(I,J)       =(I,J)-TH ELEMENT OF MATRIX A                        DFDOR
C                                                                       DFDOR
C                        ----OUTPUTS----                                DFDOR
C                                                                       DFDOR
C     DFDOM        =PARTIAL DERIVATIVE OF FPP WRT OMEGA                 DFDOR
```

```
C      DFDKX        =PARTIAL CERIVATIVE OF FPP WRT AKX
C      DFDKY        =PARTIAL CERIVATIVE OF FPP WRT AKY
C
C                    ----PROGRAM FOLLOWS BELOW----
C
       DIMENSION A(2,2),DADOM(2,2),DADKX(2,2),DADKY(2,2)
       DIMENSION RPP(2,2),DRDOM(2,2),DRDKX(2,2),DRDKY(2,2)
       GU=GI
       CALL DADOR(OMEGA,AKX,AKY,C,VX,VY,DADOM,DADKX,DADKY)
       DGDOM=(2.0*A(1,1)*DADOM(1,1)+A(1,2)*DADOM(2,1)+A(2,1)*DADOM(1,2))/
      1(2.0*GU)
       DGDKX=(2.0*A(1,1)*DADKX(1,1)+A(1,2)*DADKX(2,1)+A(2,1)*DADKX(1,2))/
      1(2.0*GU)
       DGDKY=(2.0*A(1,1)*DADKY(1,1)+A(1,2)*DADKY(2,1)+A(2,1)*DADKY(1,2))/
      1(2.0*GU)
       CALL DRDOR(OMEGA,AKX,AKY,RPP,A,DRDOM,DRDKX,DRDKY)
C      F IS R(1,1)*A(1,2)-R(1,2)*(GU+A(1,1))
       DFDOM=DRDOM(1,1)*A(1,2)-DRDOM(1,2)*(GU+A(1,1))+RPP(1,1)*DADOM(1,2)
      1-RPP(1,2)*(DGDOM+DADOM(1,1))
       DFDKX=DRDKX(1,1)*A(1,2)-DRDKX(1,2)*(GU+A(1,1))+RPP(1,1)*DADKX(1,2)
      1-RPP(1,2)*(DGDKX+DADKX(1,1))
       DFDKY=DRDKY(1,1)*A(1,2)-DRDKY(1,2)*(GU+A(1,1))+RPP(1,1)*DADKY(1,2)
      1-RPP(1,2)*(DGDKY+DADKY(1,1))
       RETURN
       END
```

```
      SUBROUTINE OMOOR(OMEGA,AKX,AKY,C,VX,VY,H,A,EM,OMOOM,OMOKX,OMOKY)   OMOOR
C     OMOOR  (SUBROUTINE)  MODIFIED  7/11/74  LAST CARD IN DECK IS NO.   OMOOR
C                                                                        OMOOR
C                                                                        OMOOR
C                     ----ABSTRACT----                                   OMOOR
C                                                                        OMOOR
C TITLE - OMOOR                                                          OMOOR
C        THE FUNCTION OF THIS SUBROUTINE IS TO COMPUTE THE COMPONENTS    OMOOR
C        OF THE MATRICES OMOOM,OMOKX, AND OMOKY WHICH REPRESENT THE      OMOOR
C        PARTIAL DERIVATIVES OF THE EM MATRIX WHICH WOULD BE             OMOOR
C        COMPUTED BY SUBROUTINE MMMM.                                    OMOOR
C              OMOOM IS THE PARTIAL DERIVATIVE MATRIX OF EM WRT OMEGA    OMOOR
C              OMOKX IS THE PARTIAL DERIVATIVE MATRIX OF EM WRT AKX      OMOOR
C              OMOKY IS THE PARTIAL DERIVATIVE MATRIX OF EM WRT AKY      OMOOR
C        MATRIX EM IS ALSO COMPUTED IN THIS SUBROUTINE.                  OMOOR
C                                                                        OMOOR
C LANGUAGE - FORTRAN V (UNIVAC 1108,REFERENECE MANUAL UP-7536 REV.1)     OMOOR
C                                                                        OMOOR
C AUTHORS - ALLAN D PIERCE,CHRISTOPHER KAPPER, G.I.T., JULY, 1974        OMOOR
C                                                                        OMOOR
C                     ----CALLING SEQUENCE----                           OMOOR
C                                                                        OMOOR
C SEE SUBROUTINE COMPK                                                   OMOOR
C     DIMENSION A(2,2),EM(2,2),OMOX(2,2)OMOOM(2,2),OMOKX(2,2)            OMOOR
C     DIMENSION OMOKY(2,2),CADOM(2,2),CADKX(2,2),DAOKY(2,2)              OMOOR
C     CALL OMOOR(OMEGA,AKX,AKY,C,VX,VY,H,A,EM,OMOOM,OMOKX,OMOKY)         OMOOR
C                                                                        OMOOR
C                     ----EXTERNAL SUBROUTINES REQUIRED----              OMOOR
C                                                                        OMOOR
C     DADOR,CAI,SAI                                                      OMOOR
C                                                                        OMOOR
C                     ----ARGUMENT LIST----                              OMOOR
C                                                                        OMOOR
C     OMEGA       R*4     NO      INP                                    OMOOR
C     AKX         R*4     NO      INP                                    OMOOR
C     AKY         R*4     NO      INP                                    OMOOR
C     C           R*4     NO      INP                                    OMOOR
C     VX          R*4     NO      INP                                    OMOOR
C     VY          R*4     NO      INP                                    OMOOR
C     H           R*4     NO      INP                                    OMOOR
C     A           R*4     2-BY-2  INP                                    OMOOR
C     EM          R*4     2-BY-2  OUT                                    OMOOR
C     OMOOM       R*4     2-BY-2  OUT                                    OMOOR
C     OMOKX       R*4     2-BY-2  OUT                                    OMOOR
C     OMOKY       R*4     2-BY-2  OUT                                    OMOOR
C                                                                        OMOOR
C NO COMMON STORAGE USED                                                 OMOOR
C                                                                        OMOOR
C                     ----INPUTS----                                     OMOOR
C                                                                        OMOOR
C     OMEGA          =ANGULAR FREQUENCY  RAD/SEC                         OMOOR
C     AKX            =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM OMOOR
C     AKY            =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM OMOOR
C     C              =SOUND SPEED IN KM/SEC                              OMOOR
C     VX             =X COMPONENT OF WIND VELOCITY IN KM/SEC             OMOOR
C     VY             =Y COMPONENT OF WIND VELOCITY IN KM/SEC             OMOOR
C     H              =THICKNESS IN KM OF LAYER                           OMOOR
C     A(I,J)         =(I,J)-TH ELEMENT OF MATRIX A FOR UHS               OMOOR
C                                                                        OMOOR
C                     ----OUTPUTS----                                    OMOOR
C                                                                        OMOOR
C     EM             =2-BY-2 TRANSFER MATRIX WHICH RELATES THE SOLUTIONS OMOOR
C                     OF THE RESIDUAL EQUATIONS AT THE TOP OF A LAYER    OMOOR
C                     TO THOSE AT THE BOTTOM OF THE LAYER.               OMOOR
```

```
C     OMDOM(I,J)  =(I,J)-TH ELEMENT OF MATRIX OMDOM                     OMDOR
C     OMDKX(I,J)  =(I,J)-TH ELEMENT OF MATRIX OMDKX                     OMDOR
C     OMDKY(I,J)  =(I,J)-TH ELEMENT OF MATRIX OMDKY                     OMDOR
C                                                                       OMDOR
C                  ----PROGRAM FOLLOWS BELOW----                        OMDOR
C                                                                       OMDOR
      DIMENSION EM(2,2),OMDX(2,2),OMDOM(2,2),OMDKX(2,2),OMDKY(2,2)      OMDOR
      DIMENSION A(2,2),DADOM(2,2),DADKX(2,2),DADKY(2,2)                 OMDOR
      CALL DADOR(OMEGA,AKX,AKY,C,VX,VY,DADOM,DADKX,DADKY)               OMDOR
      HSQ=H*H                                                           OMDOR
      X=(A(1,1)**2+A(1,2)*A(2,1))*HSQ                                   OMDOR
      CA=CAI(X)                                                         OMDOR
      SA=SAI(X)                                                         OMDOR
      DCAIX=0.5*SA                                                      OMDOR
      Y=ABS(X)                                                          OMDOR
      IF(Y-1.0E-2) 3,3,4                                                OMDOR
    3 DSAIX=1.0/6.0+X/60.0+X**2/1680.0+X**3/90720.0                     OMDOR
      GO TO 5                                                           OMDOR
    4 DSAIX=0.5*(CA-SA)/X                                               OMDOR
    5 GEM=H*DSAIX                                                       OMDOR
      DO 20 I=1,2                                                       OMDOR
      DO 20 J=1,2                                                       OMDOR
   20 OMDX(I,J)=-GEM*A(I,J)                                             OMDOR
      DO 30 I=1,2                                                       OMDOR
   30 OMDX(I,I)=OMDX(I,I)+DCAIX                                         OMDOR
      DXDOM=(2.0*A(1,1)*DADOM(1,1)+A(1,2)*DADOM(2,1)+A(2,1)*DADOM(1,2)) OMDOR
     1HSQ                                                               OMDOR
      DXDKX=(2.0*A(1,1)*DADKX(1,1)+A(1,2)*DADKX(2,1)+A(2,1)*DADKX(1,2)) OMDOR
     1HSQ                                                               OMDOR
      DXDKY=(2.0*A(1,1)*DADKY(1,1)+A(1,2)*DADKY(2,1)+A(2,1)*DADKY(1,2)) OMDOR
     1HSQ                                                               OMDOR
      T=H*SA                                                            OMDOR
      DO 90 I=1,2                                                       OMDOR
      DO 90 J=1,2                                                       OMDOR
      OMDOM(I,J)=OMDX(I,J)*DXDOM-T*DADOM(I,J)                           OMDOR
      OMDKX(I,J)=OMDX(I,J)*DXDKX-T*DADKX(I,J)                           OMDOR
      OMDKY(I,J)=OMDX(I,J)*DXDKY-T*DADKY(I,J)                           OMDOR
   90 EM(I,J)=-T*A(I,J)                                                 OMDOR
      DO 190 I=1,2                                                      OMDOR
  190 EM(I,I)=EM(I,I)+CA                                                OMDOR
      RETURN                                                            OMDOR
      END                                                               OMDOR
```

```
      SUBROUTINE ORDOR(OMEGA,AKX,AKY,RPP,A,ORDOM,ORDKX,ORDKY)
C     ORDOR  (SUBROUTINE)   MODIFIED  7/11/74  LAST CARD IN DECK IS NO.
C
C
C                  ----ABSTRACT----
C
C TITLE - ORDOR
C        THE PURPOSE OF THIS SUBROUTINE IS TO COMPUTE THE COMPONENTS
C        OF THE MATRICES ORDOM,ORDKX, AND ORDKY WHICH REPRESENT THE
C        PARTIAL DERIVATIVES OF THE RPP MATRIX WHICH WOULD BE
C        COMPUTED BY SUBROUTINE RPRP.
C             ORDOM IS THE PARTIAL DERIVATIVE MATRIX OF RPP WRT OMEGA
C             ORDKX IS THE PARTIAL DERIVATIVE MATRIX OF RPP WRT AKX
C             ORDKY IS THE PARTIAL DERIVATIVE MATRIX OF RPP WRT AKY
C
C LANGUAGE - FORTRAN V (UNIVAC 1108, REFERENCE MANUAL UP-7536 REV. 1)
C AUTHORS - A.D. PIERCE, CHRISTOPHER KAPPER, G.I.T., JULY, 1974
C
C                  ----CALLING SEQUENCE----
C
C SEE SUBROUTINE COMFK
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)
C     DIMENSION RPP(2,2),A(2,2),ORDOM(2,2),ORDKX(2,2),ORDKY(2,2)
C     DIMENSION EM(2,2),OMDOM(2,2),OMDKX(2,2),OMDKY(2,2)
C     DIMENSION UPP(2,2),DPP(2,2),AINT(2,2)
C     COMMON IMAX,CI,VXI,VYI,HI
C     CALL ORDOR(OMEGA,AKX,AKY,RPP,A,ORDOM,ORDKX,ORDKY)
C
C                  ----EXTERNAL SUBROUTINES REQUIRED----
C
C     OMDOR (OMDOR CALLS DACCR,CAI,SAI)
C
C                  ----ARGUMENT LIST----
C
C     OMEGA       R*4     NO      INP
C     AKX         R*4     NO      INP
C     AKY         R*4     NO      INP
C     RPP         R*4     2-BY-2  INP
C     A           R*4     2-BY-2  INP
C     ORDOM       R*4     2-BY-2  OUT
C     ORDKX       R*4     2-BY-2  OUT
C     ORDKY       R*4     2-BY-2  OUT
C
C COMMON STORAGE USED
C     COMMON IMAX,CI,VXI,VYI,HI
C
C     IMAX        I*4     NO      INP
C     CI          R*4     100     INP
C     VXI         R*4     100     INP
C     VYI         R*4     100     INP
C     HI          R*4     100     INP
C
C                  ----INPUTS----
C
C     OMEGA       =ANGULAR FREQUENCY  RAD/SEC
C     AKX         =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     AKY         =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     RPP         =2-BY-2 TRANSFER MATRIX WHICH CONNECTS SOLUTIONS
C                  OF THE RESIDUAL EQUATIONS AT THE BOTTOM OF THE
C                  UPPER HALFSPACE TO SOLUTIONS AT THE GROUND.
C     A           =MATRIX A OF COEFFICENTS
C
C                  ----OUTPUTS----
C
```

```
C     OROOM(I,J)  =(I,J)-TH ELEMENT OF MATRIX OROOM
C     OROKX(I,J)  =(I,J)-TH ELEMENT OF MATRIX OROKX
C     OROKY(I,J)  =(I,J)-TH ELEMENT OF MATRIX OROKY
C
C                  ----PROGRAM FOLLOWS BELLOW----
C
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      DIMENSION RPP(2,2),A(2,2),OROOM(2,2),OROKX(2,2),OROKY(2,2)
      DIMENSION EM(2,2),OMOOM(2,2),OMOKX(2,2),OMOKY(2,2)
      DIMENSION UPP(2,2),OPP(2,2),AINT(2,2)
      COMMON IMAX,CI,VXI,VYI,HI
      DO 10 I=1,2
      DO 10 J=1,2
      OROOM(I,J)=0.0
      OROKX(I,J)=0.0
   10 OROKY(I,J)=0.0
      UPP(1,1)=1.0
      UPP(1,2)=0.0
      UPP(2,1)=0.0
      UPP(2,2)=1.0
      DO 15 I=1,2
      DO 15 J=1,2
   15 OPP(I,J)=RPP(I,J)
      DO 100 J=1,IMAX
      I=IMAX+1-J
      C=CI(I)
      VX=VXI(I)
      VY=VYI(I)
      H=HI(I)
      CALL OMOOR(OMEGA,AKX,AKY,C,VX,VY,H,A,EM,OMOOM,OMOKX,OMOKY)
      MULTIPLY OPP TIMES THE INVERSE OF EM
      AINT(1,1)=OPP(1,1)*EM(2,2)-OPP(1,2)*EM(2,1)
      AINT(1,2)=-OPP(1,1)*EM(1,2)+OPP(1,2)*EM(1,1)
      AINT(2,1)=OPP(2,1)*EM(2,2)-OPP(2,2)*EM(2,1)
      AINT(2,2)=-OPP(2,1)*EM(1,2)+OPP(2,2)*EM(1,1)
      DO 20 II=1,2
      DO 20 JJ=1,2
   20 OPP(II,JJ)=AINT(II,JJ)
      DO 30 II=1,2
      DO 30 JJ=1,2
      DO 30 KK=1,2
      DO 30 LL=1,2
      OROOM(II,JJ)=OROOM(II,JJ)+OPP(II,KK)*OMOOM(KK,LL)*UPP(LL,JJ)
      OROKX(II,JJ)=OROKX(II,JJ)+OPP(II,KK)*OMOKX(KK,LL)*UPP(LL,JJ)
   30 OROKY(II,JJ)=OROKY(II,JJ)+OPP(II,KK)*OMOKY(KK,LL)*UPP(LL,JJ)
      DO 40 II=1,2
      DO 40 JJ=1,2
   40 AINT(II,JJ)=EM(II,1)*UPP(1,JJ)+EM(II,2)*UPP(2,JJ)
      DO 50 II=1,2
      DO 50 JJ=1,2
   50 UPP(II,JJ)=AINT(II,JJ)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ELINT(OMEGA,AKX,AKY,C,VX,VY,H,F1H,F2H,A1,A2,AINT)      ELINT
C     ELINT (SUBROUTINE)                  7/25/68   LAST CARD IN DECK IS ELINT
C                                                                       ELINT
C                  ----ABSTRACT----                                     ELINT
C                                                                       ELINT
C TITLE - ELINT                                                         ELINT
C          THIS SUBROUTINE COMPUTES THE INTEGRAL                        ELINT
C                                                                       ELINT
C            AINT = INTEGRAL OVER Z FROM 0 TO H OF                      ELINT
C                                                                       ELINT
C                  (A1*F1(Z) + A2*F2(Z))**2                          (1 ELINT
C                                                                       ELINT
C          THE FUNCTIONS F1(Z) AND F2(Z) ARE THE SOLUTIONS OF THE COUPLED ELINT
C          ORDINARY DIFFERENTIAL EQUATIONS                              ELINT
C                                                                       ELINT
C            DF1/DZ = A11*F1 + A12*F2                                (2 ELINT
C            DF2/DZ = A21*F1 + A22*F2                                (2 ELINT
C                                                                       ELINT
C          WHERE THE ELEMENTS OF THE MATRIX A ARE INDEPENDENT OF Z.     ELINT
C          FOR GIVEN SOUND SPEED C, WIND VELOCITY COMPONENTS VX AND VY,  ELINT
C          ANGULAR FREQUENCY OMEGA, AND WAVE NUMBER COMPONENTS AKX AND AK ELINT
C          THE A(I,J) ARE COMPUTED BY CALLING AAAA.  THE SOLUTION TO THE ELINT
C          DIFFERENTIAL EQUATIONS IS FIXED BY SPECIFICATION OF F1 AND F2 ELINT
C          AT Z=H.                                                      ELINT
C                                                                       ELINT
C PROGRAM NOTES                                                         ELINT
C                                                                       ELINT
C                THE GENERAL SOLUTION OF EQNS. (2) IS                   ELINT
C                                                                       ELINT
C                    F1(Z) = CAI(X)*F1(H)-(H-Z)*SAI(X)*(A11*F1(H)+A12*F2(H ELINT
C                    F2(Z) = CAI(X)*F2(H)-(H-Z)*SAI(X)*(A21*F1(H)+A22*F2(H ELINT
C                                                                       ELINT
C                WITH X=(A11**2+A12*A21)*(H-Z)**2 SINCE A22=-A11.  WE LET ELINT
C                                                                       ELINT
C                    R1 =(INTEGRAL OF (CAI(X))**2)*(2/H)                ELINT
C                    R2 =(INTEGRAL OF ((H-Z)*SAI(X))**2)*(2/H**3)       ELINT
C                    R3 =(INTEGRAL OF ((H-Z)*SAI(X)*CAI(X)))*(4/H**2)   ELINT
C                                                                       ELINT
C                WHERE IN EACH CASE THE INTEGRATION IS OVER Z FROM 0 TO H ELINT
C                THE QUANTITIES R1,R2,R3 ARE COMPUTED BY CALLING BBBB.  ELINT
C                THEN                                                   ELINT
C                                                                       ELINT
C                    AINT=(H/2)*(FP1)**2*R1+(H**3/2)*(FP2)**2*R2        ELINT
C                         -(H**2/2)*(FP1)*(FP2)*R3                      ELINT
C                                                                       ELINT
C                WITH                                                   ELINT
C                    FP1= A1*F1(H)+A2*F2(H)                             ELINT
C                    FP2= A1*(A11*F1(H)+A12*F2(H))+A2*(A21*F1(H)+A22*F2(H) ELINT
C                                                                       ELINT
C                THE LATTER TWO QUANTITIES REPRESENT THE COEFFICIENTS OF ELINT
C                CAI(X) AND (H-Z)*SAI(X) IN A1*F1+A2*F2.                ELINT
C                                                                       ELINT
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)            ELINT
C                                                                       ELINT
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968                            ELINT
C                                                                       ELINT
C                 ----CALLING SEQUENCE----                              ELINT
C                                                                       ELINT
C SEE SUBROUTINE TOTINT                                                 ELINT
C NO DIMENSION STATEMENTS REQUIRED                                      ELINT
C     CALL ELINT(OMEGA,AKX,AKY,C,VX,VY,H,F1H,F2H,A1,A2,AINT)            ELINT
C                                                                       ELINT
C                 ----EXTERNAL SUBROUTINES REQUIRED----                 ELINT
C                                                                       ELINT
```

```
C     AAAA, BBBB                                                        ELINT
C                                                                       ELINT
C                     ----ARGUMENT LIST----                             ELINT
C                                                                       ELINT
C     OMEGA       R*4     NO     INP                                    ELINT
C     AKX         R*4     NO     INP                                    ELINT
C     AKY         R*4     NO     INP                                    ELINT
C     C           R*4     NO     INP                                    ELINT
C     VX          R*4     NO     INP                                    ELINT
C     VY          R*4     NO     INP                                    ELINT
C     H           R*4     NO     INP                                    ELINT
C     F1H         R*4     NO     INP                                    ELINT
C     F2H         R*4     NO     INP                                    ELINT
C     A1          R*4     NO     INP                                    ELINT
C     A2          R*4     NO     INP                                    ELINT
C     AINT        R*4     NO     OUT                                    ELINT
C                                                                       ELINT
C NO COMMON STORAGE USED                                                ELINT
C                                                                       ELINT
C                     ----INPUTS----                                    ELINT
C                                                                       ELINT
C     OMEGA       =ANGULAR FREQUENCY IN RADIANS/SEC                     ELINT
C     AKX         =X COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)        ELINT
C     AKY         =Y COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)        ELINT
C     C           =SOUND SPEED IN KM/SEC                                ELINT
C     VX          =X COMPONENT OF WIND VELOCITY IN KM/SEC               ELINT
C     VY          =Y COMPONENT OF WIND VELOCITY IN KM/SEC               ELINT
C     H           =INTEGRATION INTERVAL (LAYER THICKNESS) IN KM         ELINT
C     F1H         =VALUE OF F1(Z) AT UPPER LIMIT OF INTEGRAL            ELINT
C     F2H         =VALUE OF F2(Z) AT UPPER LIMIT OF INTEGRAL            ELINT
C     A1          =COEFFICIENT OF F1(Z) IN INTEGRAND                    ELINT
C     A2          =COEFFICIENT OF F2(Z) IN INTEGRAND                    ELINT
C                                                                       ELINT
C                     ----OUTPUTS----                                   ELINT
C                                                                       ELINT
C     AINT        =INTEGRAL OVER HEIGHT WITH RANGE H OF THE QUANTITY    ELINT
C                  (A1*F1(Z)+A2*F2(Z))**2 WHERE F1(Z) AND F2(Z) ARE     ELINT
C                  EQUAL TO F1H AND F2H, RESPECTIVELY, AT THE UPPER     ELINT
C                  LIMIT AND SATISFY THE RESIDUAL DIFFERENTIAL EQUATION ELINT
C                                                                       ELINT
C                     ----PROGRAM FOLLOWS BELOW----                     ELINT
C                                                                       ELINT
      DIMENSION A(2,2)                                                  ELINT
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)                                ELINT
C                                                                       ELINT
C COMPUTATION OF FP1 AND FP2                                            ELINT
      FP1=A1*F1H+A2*F2H                                                 ELINT
      FP2=A1*(A(1,1)*F1H+A(1,2)*F2H)+A2*(A(2,1)*F1H+A(2,2)*F2H)         ELINT
C                                                                       ELINT
C COMPUTATION OF COEFFICIENTS OF R1,R2,R3                               ELINT
      S1=0.5*H*FP1**2                                                   ELINT
      S2=0.5*(H**3)*FP2**2                                              ELINT
      S3=-0.5*(H**2)*FP1*FP2                                            ELINT
C                                                                       ELINT
C COMPUTATION OF R1,R2,R3                                               ELINT
      X=(A(1,1)**2+A(1,2)*A(2,1))*H**2                                  ELINT
      CALL BBBB(X,R1,R2,R3)                                             ELINT
C                                                                       ELINT
C COMPUTATION OF AINT                                                   ELINT
      AINT=S1*R1+S2*R2+S3*R3                                            ELINT
      RETURN                                                            ELINT
      END                                                               ELINT
```

```
      FUNCTION FNMO01(V)                                                FNMO01
C     FNMO01 (FUNCTION)                   6/19/68   LAST CARD IN DECK IS FNMO01
C                                                                       FNMO01
C                                                                       FNMO01
C                      ----ABSTRACT----                                 FNMO01
C                                                                       FNMO01
C TITLE - FNMO01                                                        FNMO01
C           EVALUATATION OF NORMAL MODE DISPERSION FUNCTION AS FUNCTION OF FNMO01
C           PHASE VELOCITY V                                            FNMO01
C                                                                       FNMO01
C                  THE NORMAL MODE DISPERSION FUNCTION DEPENDS ON THREE VAR FNMO01
C                  ABLES, ANGULAR FREQUENCY OMEGA, PHASE VELOCITY V, AND   FNMO01
C                  DIRECTION OF PROPAGATION THETK.  FNMO01 OBTAINS V THROUG FNMO01
C                  ITS ARGUMENT, OMEGA AND THETK FROM COMMON.  SUBROUTINE  FNMO01
C                  NMOFN IS THEN CALLED TO EVALUATE THE FUNCTION.  (SEE   FNMO01
C                  PIERCE, J.COMP.PHYSICS, FEB.,1967, P.343-366 FOR DEFINI- FNMO01
C                  TION OF NORMAL MODE DISPERSION FUNCTION.)              FNMO01
C                                                                       FNMO01
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             FNMO01
C                                                                       FNMO01
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                FNMO01
C                                                                       FNMO01
C                                                                       FNMO01
C                      ----USAGE----                                    FNMO01
C                                                                       FNMO01
C     OMEGA MUST BE STORED IN WORD POSITION 402 OF UNLABELED COMMON, AN FNMO01
C     THETK MUST BE IN POSITION 404.                                    FNMO01
C                                                                       FNMO01
C     FNMO01 CALLS SUBROUTINE NMOFN WHICH CALLS AAAA AND RRRR.  RRRR     FNMO01
C     CALLS AAAA AND MMMM.  ALL THESE SUBROUTINES ARE DESCRIBED ELSE-    FNMO01
C     WHERE IN THIS SERIES.                                             FNMO01
C                                                                       FNMO01
C CALLING SEQUENCE                                                      FNMO01
C                                                                       FNMO01
C     COMMON CM1(401),OMEGA,CM2,THETK                                   FNMO01
C     OMEGA = XXX                                                       FNMO01
C     THETK = XXX                                                       FNMO01
C     V = XXX                                                           FNMO01
C     FUNCTN = FNMO01(V)                                                FNMO01
C                                                                       FNMO01
C INPUTS                                                                FNMO01
C                                                                       FNMO01
C    V           PHASE VELOCITY (KM/SEC).                               FNMO01
C      R*4                                                              FNMO01
C                                                                       FNMO01
C    OMEGA       ANGULAR FREQUENCY (RADIANS/SEC).                       FNMO01
C      R*4                                                              FNMO01
C                                                                       FNMO01
C    THETK       PHASE VELOCITY DIRECTION MEASURED COUNTER-CLOCKWISE FROM FNMO01
C      R*4       X-AXIS.                                                FNMO01
C                                                                       FNMO01
C OUTPUTS                                                               FNMO01
C                                                                       FNMO01
C    THE ONLY OUTPUT IS THE VALUE OF THE NORMAL MODE DISPERSION FUNCTIO FNMO01
C    FOR THE VALUES OF V, OMEGA, AND THETK WHICH HAVE BEEN INPUT.       FNMO01
C                                                                       FNMO01
C                                                                       FNMO01
C                      ----PROGRAM FOLLOWS BELOW----                    FNMO01
C                                                                       FNMO01
C                                                                       FNMO01
C                                                                       FNMO01
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)                       FNMO01
      COMMON IMAX,CI,VXI,VYI,HI,OMEGAC,VPHSEC,THETK _                   FNMO01
C                                                                       FNMO01
C OMEGA AND THETK OBTAINED FROM COMMON                                  FNMO01
      OMEGA=OMEGAC                                                      FNMO01
      CALL NMOFN(OMEGA,V,THETK,L,FPP,K)                                 FNMO01
      FNMO01=FPP                                                        FNMO01
      RETURN                                                            FNMO01
      END                                                               FNMO01
```

```
      SUBROUTINE LNGTHN(OM,V,INMODE,NOM,NVP,NVPP,N1,KL,THETK)           LNGTHN
C     LNGTHN (SUBROUTINE)              7/19/68   LAST CARD IN DECK IS LNGTHN
C                                                                       LNGTHN
C                                                                       LNGTHN
C                         ----ABSTRACT----                              LNGTHN
C                                                                       LNGTHN
C TITLE - LNGTHN                                                        LNGTHN
C        LENGTHEN THE MATRIX INMODE BY ADDING KL ROWS BETWEEN THE N1 AN LNGTHN
C        N1+1 ROWS                                                      LNGTHN
C                                                                       LNGTHN
C                 LNGTHN ADDS KL ELEMENTS TO THE VECTOR OF PHASE VELOCITIE LNGTHN
C                 V , DIVIDING THE INTERVAL BETWEEN V(N1) AND V(N1+1) INTO LNGTHN
C                 KL+1 EQUAL PARTS.  FOR EACH NEW PHASE VELOCITY, A NEW RO LNGTHN
C                 IS ADDED TO THE INMODE MATRIX (DEFINED IN SUBROUTINE    LNGTHN
C                 MPOUT).  INMODE IS STORED COLUMN BY COLUMN IN VECTOR FOR LNGTHN
C                                                                       LNGTHN
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)           LNGTHN
C AUTHOR      - J.W.POSEY, M.I.T., JUNE,1968                            LNGTHN
C                                                                       LNGTHN
C                                                                       LNGTHN
C                         ----USAGE----                                 LNGTHN
C                                                                       LNGTHN
C     OM,V,INMODE MUST BE DIMENSIONED IN THE CALLING PROGRAM            LNGTHN
C     NMOFN IS ONLY SUBROUTINE CALLED                                   LNGTHN
C                                                                       LNGTHN
C FORTRAN USAGE                                                         LNGTHN
C     CALL LNGTHN(OM,V,INMODE,NOM,NVP,NVPP,N1,KL,THETK)                 LNGTHN
C                                                                       LNGTHN
C INPUTS                                                                LNGTHN
C                                                                       LNGTHN
C    OM        VECTOR WHOSE ELEMENTS ARE THE VALUES OF ANGULAR FRE-     LNGTHN
C      R*4(D)  QUENCY CORRESPONDING TO THE COLUMNS OF THE INMODE MATRIX LNGTHN
C                                                                       LNGTHN
C    V         VECTOR WHOSE ELEMENTS ARE THE VALUES OF PHASE VELOCITY   LNGTHN
C      R*4(D)  CORRESPONDING TO THE ROWS OF THE INMODE MATRIX.          LNGTHN
C                                                                       LNGTHN
C    INMODE    EACH ELEMENT OF THIS MATRIX CORRESPONDS TO A POINT IN TH LNGTHN
C      I*4(D)  FREQUENCY (OM) - PHASE VELOCITY (V) PLANE.  IF THE NORMA LNGTHN
C              MODE DISPERSION FUNCTION (FPP) IS POSITIVE AT THAT POINT LNGTHN
C              THE ELEMENT IS +1.  IF FPP IS NEGATIVE, THE ELEMENT IS   LNGTHN
C              -1.  IF FPP DOES NOT EXIST, THE ELEMENT IS 5.  INMODE HA LNGTHN
C              NVP ROWS (INCREASED TO NVPP) AND NOM COLUMNS.  MATRIX IS LNGTHN
C              STORED IN VECTOR FORM COLUMN AFTER COLUMN.               LNGTHN
C    NOM       THE NUMBER OF ELEMENTS IN OM.                            LNGTHN
C      I*4                                                              LNGTHN
C    NVP       THE NUMBER OF ELEMENTS IN V (WHEN LNGTHN IS CALLED).     LNGTHN
C      I*4                                                              LNGTHN
C    N1        NUMBER OF INMODE ROW IMMEDIATELY ABOVE SPACE IN WHICH NE LNGTHN
C      I*4     ROWS ARE TO BE ADDED                                     LNGTHN
C                                                                       LNGTHN
C    KL        NUMBER OF ROWS TO BE ADDED                               LNGTHN
C      I*4                                                              LNGTHN
C    THETK     PHASE VELOCITY DIRECTION (RADIANS)                       LNGTHN
C      R*4                                                              LNGTHN
C                                                                       LNGTHN
C OUTPUTS                                                               LNGTHN
C                                                                       LNGTHN
C    THE OUTPUTS ARE NVPP (= NVP + KL) AND REVISED VERSIONS OF V AND    LNGTHN
C    INMODE.                                                            LNGTHN
C                                                                       LNGTHN
C                                                                       LNGTHN
C                         ----EXAMPLE----                               LNGTHN
C                                                                       LNGTHN
C   VALUES OF INMODE NOT VALID -- FOR ILLUSTRATION PURPOSES ONLY        LNGTHN
C                                                                       LNGTHN
C              V=1.0,2.0                                                LNGTHN
C              OM=1.0,2.0                                               LNGTHN
C              INMODE=1,-1,-1,1                                         LNGTHN
```

```
C              CALL LNGTHN(OM,V,INMODE,2,2,NVPP,1,3,THETK)
C
C         UPON RETURN TO CALLING PROGRAM THE VALUES OF V AND NVPP ARE
C              V=1.0,1.25,1.5,1.75,2.0
C              NVPP=5
C         INMODE WILL BE OF THE FORM
C              INMODE=1,Y,Y,Y,-1,-1,Y,Y,Y,1
C         WHERE THE Y"S ARE NEW ELEMENTS, EACH OF WHICH MAY BE -1, 1,
C         OR 5
C
C         ORIGINAL MATRIX                  EXPANDED MATRIX
C
C                  +-                             +-
C                  -+                             YY
C                                                 YY
C                                                 YY
C                                                 -+
C
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C
C VARIABLE DIMENSIONING
      DIMENSION OM(1),V(1),INMODE(1)
      COMMON IMAX,CI(100),VXI(100),VYI(100),HI(100)
      DELVP = (V(N1+1)-V(N1)) / (KL+1)
C DELPP IS THE INTERVAL OF PHAS VELOCITIES FOR THE ADDED ROWS.
      NVPP = NVP + KL
C NVPP IS THE NEW NUMBER OF ROWS IN THE TOTAL MATRIX.
C
C N2 IS NEW NUMBER OF OLD ROW NO. (N1+1)
      N2 = N1 + KL + 1
C
C SHIFT OLD VALUES OF V(I) IN LOWER ROWS TO I+KL SPOTS  ONE HAS TO
C SHIFT THE NVP ELEMENT FIRST.  NOTE THAT I RANGES FROM NVPP TO N2
C DOWNWARD WHILE I-KL RANGES FROM NVP TO N1+1.
      DO 71 IP =N2,NVPP
      I = NVPP - (IP-N2)
   71 V(I) = V(I-KL)
C
C NEW VALUES OF VP ARE INSERTED INTO V
      DO 72 IP=1,KL
      I = N1 + IP
   72 V(I) = V(N1) + IP*DELVP
C
C BEGINNING AT THE RIGHT INMODE IS LENGTHENED COLUMN BY COLUMN
      DO 90 JP=1,NOM
      J = NOM - (JP-1)
      DO 90 IP=1,NVPP
      I = NVPP- (IP-1)
C
C THE IJ ELEMENT IN THE INMODE VECTOR IS THE J ELEMENT IN THE I ROW OF
C THE NEW INMODE MATRIX
      IJ = (J-1)*NVPP + I
C
C IF I CORRESPONDS TO A NEW ROW INMODE(IJ) MUST BE DETERMINED FROM NMOF
      IF (I.GT.N1.AND.I.LT.N2) GO TO 9
C
C IJOLD IS NO. OF ELEMENT IN OLD INMODE VECTOR WHICH IS TO BE MOVED INT
C IJ POSITION OF NEW VECTOR
      IJOLD = (J-1)*NVP + I
C NOTE THAT IOLD IS ALWAYS I IF I .LT. N1 BUT IOLD IS I-KL IF I .GE. N2
C IJOLD IS COMPUTED ON THE BASIS OF NVP RATHER THAN NVPP ROWS.
```

```
      IF (I.GE.N2) IJOLD = IJOLD - KL
      INMODE(IJ) = INMODE(IJOLD)
      GO TO 80
C
    9 CALL NMOFN(OM(J),V(I),THETK,L,FPP,K)
C
C IF FPP EXISTS L = 1 AND INMODE(IJ) = (FPP/ABS(FPP))
      INMODE(IJ) = 1
      IF (L.EQ.1.AND.FFP.LE.0.0) INMODE(IJ) = -1
C
C IF FPP DOES NOT EXIST L = -1
      IF (L.EQ.-1) INMODE(IJ)=5
C
   80 CONTINUE
   90 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MMMM(OMEGA,AKX,AKY,C,VX,VY,H,EM)                       MMMM
C     MMMM (SUBROUTINE)                   7/25/68   LAST CARD IN DECK IS MMMM
C                                                                       MMMM
C                    ----ABSTRACT----                                   MMMM
C                                                                       MMMM
C TITLE - MMMM                                                          MMMM
C        THIS SUBROUTINE COMPUTES THE 2-BY-2 TRANSFER MATRIX EM WHICH   MMMM
C        CONNECTS THE SOLUTIONS OF THE RESIDUAL EQUATIONS AT THE TOP    MMMM
C        OF A LAYER TO THOSE AT THE BOTTOM OF THE LAYER BY THE RELATION MMMM
C                                                                       MMMM
C           PHI1(ZB)= EM(1,1)*PHI1(ZB+H)+ EM(1,2)*PHI2(ZB+H)            MMMM
C                                                                       MMMM
C           PHI2(ZB)= EM(2,1)*PHI1(ZB+H)+ EM(2,2)*PHI2(ZB+H)            MMMM
C                                                                       MMMM
C        WHERE ZB DENOTES THE HEIGHT OF THE BOTTOM OF AN ISOTHERMAL     MMMM
C        LAYER (THICKNESS H) WITH CONSTANT WINDS.  THE QUANTITIES       MMMM
C        PHI1(Z) AND PHI2(Z) SATISFY THE RESIDUAL EQUATIONS.            MMMM
C                                                                       MMMM
C           D(PHI1)/DZ = A(1,1)*PHI1(Z) + A(1,2)*PHI2(Z)                MMMM
C                                                                       MMMM
C           D(PHI2)/DZ = A(2,1)*PHI1(Z) + A(2,2)*PHI2(Z)                MMMM
C                                                                       MMMM
C        WHERE THE A(I,J) ARE CONSTANT OVER THE LAYER AND WHERE         MMMM
C        A(2,2)=-A(1,1).  ON THIS BASIS, ONE CAN SHOW THAT              MMMM
C                                                                       MMMM
C           EM(I,J) =CAI(X)*KDELTA(I,J)-H*SAI(X)*A(I,J)                 MMMM
C                                                                       MMMM
C        WHERE                                                          MMMM
C                 X =(A(1,1)**2+A(1,2)*A(2,1))*H**2                     MMMM
C                                                                       MMMM
C        AND WHERE KDELTA(I,J) IS THE KRONECKER DELTA (1 IF INDICES     MMMM
C        EQUAL, 0 OTHERWISE).  THE FUNCTIONS CAI AND SAI ARE DEFINED IN MMMM
C        THE DESCRIPTIONS OF THE CORRESPONDING FUNCTION SUBPROGRAMS.    MMMM
C                                                                       MMMM
C        THE MATRIX A IS COMPUTED FOR GIVEN FREQUENCY, WAVE NUMBER, SOU MMMM
C        SPEED, AND WIND VELOCITY BY CALLING SUBROUTINE AAAA.           MMMM
C                                                                       MMMM
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)            MMMM
C                                                                       MMMM
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968.                           MMMM
C                                                                       MMMM
C                    ----CALLING SEQUENCE----                           MMMM
C                                                                       MMMM
C SEE SUBROUTINES NAPPCE,RRRR                                           MMMM
C     DIMENSION EM(2,2)                                                 MMMM
C     CALL MMMM(OMEGA,AKX,AKY,C,VX,VY,H,EM)                             MMMM
C                                                                       MMMM
C                    ----EXTERNAL SUBROUTINES REQUIRED----              MMMM
C                                                                       MMMM
C     AAAA,CAI,SAI                                                      MMMM
C                                                                       MMMM
C                    ----ARGUMENT LIST----                              MMMM
C                                                                       MMMM
C     OMEGA      R*4     NO     INP                                     MMMM
C     AKX        R*4     NO     INP                                     MMMM
C     AKY        R*4     NO     INP                                     MMMM
C     C          R*4     NO     INP                                     MMMM
C     VX         R*4     NO     INP                                     MMMM
C     VY         R*4     NO     INP                                     MMMM
C     A          R*4     NO     INP                                     MMMM
C     EM         R*4    2-BY-2 OUT                                      MMMM
C                                                                       MMMM
C NO COMMON STORAGE IS USED                                             MMMM
C                                                                       MMMM
C
```

```
C                       ----INPUTS----                                   MHMM
C                                                                        MHMM
C     OMEGA            =ANGULAR FRECUENCY IN RAC/SEC                     MHMM
C     AKX              =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM MHMM
C     AKY              =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM MHMM
C     C                =SOUND SPEED IN KM/SEC                            MHMM
C     VX               =X COMPONENT OF WINC VELOCITY IN KM/SEC           MHMM
C     VY               =Y COMPONENT OF WINC VELOCITY IN KM/SEC           MHMM
C     H                =THICKNESS IN KM OF LAYER                         MHMM
C                                                                        MHMM
C                       ----OUTPUTS----                                  MHMM
C                                                                        MHMM
C     EM               =2-BY-2 TRANSFER MATRIX WHICH RELATES THE SOLUTIONS O MHMM
C                       THE RESIDUAL EQUATICNS AT THE TOP OF A LAYER TO THOS MHMM
C                       AT THE BOTTOM OF THE LAYER                       MHMM
C                                                                        MHMM
C                       ----PROGRAM FOLLOWS BELOW----                    MHMM
C                                                                        MHMM
C                                                                        MHMM
      DIMENSION A(2,2),EM(2,2)                                           MHMM
C                                                                        MHMM
C COMPUTE A(I,J), CAI(X), ANC SAI(X)                                     MHMM
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)                                 MHMM
      X=(A(1,1)**2+A(1,2)*A(2,1))*H**2                                   MHMM
      CA=CAI(X)                                                          MHMM
      SA=SAI(X)                                                          MHMM
C                                                                        MHMM
C COMPUTE THE TERMS -H*SAI(X)*A(I,J)                                     MHMM
      TA=H*SA                                                            MHMM
      DO 90 I=1,2                                                        MHMM
      DO 90 J=1,2                                                        MHMM
   90 EM(I,J)=-TA*A(I,J)                                                 MHMM
C                                                                        MHMM
C ADD IN CAI(X)*KDELTA(I,J) TERMS BY ADDING CA TO DIAGONAL ELEMENTS      MHMM
      DO 190 I=1,2                                                       MHMM
  190 EM(I,I)=EM(I,I)+CA                                                 MHMM
C                                                                        MHMM
      RETURN                                                             MHMM
      END                                                                MHMM
```

```
      SUBROUTINE MODETR(IST,JST,NMODE,KST,KFIN,OMMOD,VPMOD,NROW,NCOL,   MODETR
     1 INMODE,OM,VP,KRUD)                                                MODETR
C     MODETR (SUBROUTINE)              6/25/68   LAST CARD IN DECK IS MODETR
C                                                                       MODETR
C                                                                       MODETR
C                      ----ABSTRACT----                                 MODETR
C                                                                       MODETR
C TITLE - MODETR                                                        MODETR
C           PROGRAM TO TABULATE A TABLE OF PHASE VELOCITY VERSUS FREQUENCY MODETR
C           FOR A GIVEN GUIDED MODE.  THE NORMAL MODE DISPERSION FUNCTION MODETR
C           IS ZERO FOR EACH LISTING OF THE TABLE.  THE COMPUTATIONAL    MODETR
C           METHOD IS BASED ON THE PREVIOUSLY COMPUTED VALUES OF THE NMDF MODETR
C           SIGN INMODE((J-1)*NROW+I) AT POINTS (I,J) IN A RECTANGULAR   MODETR
C           ARRAY OF NROW ROWS AND NCOL COLUMNS.  DIFFERENT COLUMNS (J)  MODETR
C           CORRESPOND TO DIFFERENT FREQUENCIES WHILE DIFFERENT ROWS (I) MODETR
C           CORRESPOND TO DIFFERENT PHASE VELOCITIES.  DISPERSION CURVES MODETR
C           OF VARIOUS MODES APPEAR ON THIS ARRAY AS LINES OF DEMARCATION MODETR
C           BETWEEN ADJACENT REGIONS WITH DIFFERENT INMODES.  TWO ADJACENT MODETR
C           POINTS WITH INMODES OF OPPOSITE SIGN BRACKET A POINT ON THE  MODETR
C           ACTUAL DISPERSION CURVE.  IF THE POINTS CORRESPOND TO THE SAME MODETR
C           FREQUENCY, THEN THE PHASE VELOCITY CORRESPONDING TO THAT OMEGA MODETR
C           ON THE DISPERSION CURVE IS FOUND BY CALLING RTMI, A 360 PACKAG MODETR
C           ROUTINE FOR SOLVING NONLINEAR EQUATIONS, AND CONSIDERING THE MODETR
C           NMDF AS A FUNCTION OF VPHSE WITH OMEGA FIXED.  SIMILARLY, IF MODETR
C           THE POINTS CORRESPOND TO THE SAME PHASE VELOCITY, THE APPROPRI MODETR
C           OMEGA CORRESPONDING TO THIS PHASE VELOCITY IS FOUND BY CALLING MODETR
C           RTMI WITH THE NMDF CONSIDERED AS A FUNCTION OF OMEGA WITH    MODETR
C           VPHSE FIXED.                                                 MODETR
C                                                                       MODETR
C           THE PROGRAM SUCCESSIVELY CONSIDERS EACH PAIR OF ADJACENT POINT MODETR
C           WITH OPPOSITE INMODES BRACKETING A LINE OF DEMARCATION AND   MODETR
C           PROCEEDS IN THE DIRECTION OF INCREASING FREQUENCY UNDER THE  MODETR
C           ASSUMPTION THAT THE PHASE VELOCITY CURVE SLOPES DOWNWARDS.   MODETR
C                                                                       MODETR
C PROGRAM NOTES                                                         MODETR
C                                                                       MODETR
C                THE MODES ARE NUMBERED.  THE INPUT INTEGER NMODE DESIGNA MODETR
C                WHICH MODE IS BEING TABULATED.  THE PAIRS OF FREQUENCY  MODETR
C                AND PHASE VELOCITY VALUES ARE STORED AS OMMOD(KST(NMODE)) MODETR
C                OMMOD(KST(NMODE)+1),OMMOD(KST(NMODE)+2),...............  MODETR
C                OMMOD(KFIN(NMODE)),VPMOD(KST(NMODE)),VPMOD(KST(NMODE)+1) MODETR
C                ......VPMOD(KFIN(NMODE)).  THE ARRAYS OMMOD AND VPMOD   MODETR
C                ARE USED TO STORE DISPERSION CURVES FOR ALL MODES.      MODETR
C                                                                       MODETR
C                KST(NMODE) IS INPUT WHILE KFIN(NMODE) IS DETERMINED DURI MODETR
C                THE COMPUTATION.  THE TOTAL NUMBER OF POINTS EXTRACTABLE MODETR
C                FROM THE ARRAY OF INMODE VALUES DETERMINES KFIN-KST+1.  MODETR
C                IF A SINGLE POINT CANNOT BE CALCULATED, THE PROGRAM     MODETR
C                RETURNS KRUD=-1.  OTHERWISE IT RETURNS KRUD=1.          MODETR
C                                                                       MODETR
C                THE SUBROUTINE RTMI FOR SOLVING A NONLINEAR EQUATION    MODETR
C                IS ALLOWED A MAXIMUM OF TEN ITERATIONS TO FIND THE      MODETR
C                PHASE VELOCITY TO ACCURACY OF 1.E-5 KM/SEC OR THE       MODETR
C                FREQUENCY TO FOUR SIGNIFICANT FIGURES.  IF THE SEARCH IS MODETR
C                UNSUCCESSFUL A MESSAGE IS PRINTED AND THE POINT IS      MODETR
C                SKIPPED OVER.                                           MODETR
C                                                                       MODETR
C                THE INPUT PARAMETERS IST,JST ARE COORDINATES OF A POINT MODETR
C                THE INMODE ARRAY.  THIS POINT SHOULD BE THAT POINT FURTH MODETR
C                TO THE UPPER LEFT OF THOSE POINTS LYING BELOW THE LINE O MODETR
C                DEMARCATION FOR THE MODE CONSIDERED, PROVIDING THAT POIN MODETR
C                DOES NOT HAVE INMODE=5.                                 MODETR
C                                                                       MODETR
C LANGUAGE  - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)             MODETR
```

```
C AUTHOR     - A.D.PIERCE, M.I.T., JUNE,1968                         MODETR
C                                                                    MODETR
C                   ----CALLING SEQUENCE----                         MODETR
C                                                                    MODETR
C SEE SUBROUTINE ALLMOD                                              MODETR
C     DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),INMODE(1),OM(1),VP(1)  MODETR
C               (SUBROUTINE USES VARIABLE DIMENSIONING)              MODETR
C     CALL MODETR(IST,JST,NMODE,KST,KFIN,OMMOD,VPMOD,NROW,NCOL,INMODE,  MODETR
C    1 OM,VP,KRUD)                                                   MODETR
C     IF( KRUD .EQ. 1 ) GO SOMEWHERE                                 MODETR
C                                                                    MODETR
C                   ----EXTERNAL SUBROUTINES REQUIRED----            MODETR
C                                                                    MODETR
C     NXTPNT, RTMI, FNMOD1, FNMOD2, NMOFN, AAAA, RRRR, MMMM,CAI,SAI   MODETR
C       (FNMOD1 AND FNMODE CALL NMOFN, WHICH IN TURN CALLS AAAA AND RRR MODETR
C       RRRR CALLS AAAA AND MMMM.  DESCRIPTIONS OF THESE PROGRAMS ARE  MODETR
C       GIVEN ELSEWHERE IN THIS SERIES.)                             MODETR
C                                                                    MODETR
C     RTMI IS A SUBROUTINE CODED BY IBM TO DETERMINE A ROOT OF A GENERA MODETR
C     NONLINEAR EQUATION F(X)=0 BY MEANS OF MUELLER-S ITERATION SCHEME  MODETR
C     OF SUCCESSIVE BISECTION AND INVERSE PARABOLIC INTERPOLATION.  A  MODETR
C     COMPLETE DESCRIPTION AND DECK LISTING IS GIVEN ON PAGES 198-199 O MODETR
C     DOCUMENT H20-0205-2, SYSTEM/360 SCIENTIFIC SUBROUTINE PACKAGE   MODETR
C     (360A-CM-03X) VERSION II, PROGRAMMER-S MANUAL, IBM, TECHNICAL    MODETR
C     PUBLICATIONS DEPARTMENT, 112 EAST POST ROAD, WHITE PLAINS, N.Y.  MODETR
C     10601, PUBLISHED 1966, 1967.                                   MODETR
C                                                                    MODETR
C                ----ARGUMENT LIST----                               MODETR
C                                                                    MODETR
C     IST        I*4    NO    INP                                    MODETR
C     JST        I*4    NO    INP                                    MODETR
C     NMODE      I*4    NO    INP                                    MODETR
C     KST        I*4    VAR   INP (ONLY KST(NMODE) NEEDED)           MODETR
C     KFIN       I*4    VAR   OUT (ONLY KFIN(NMODE) COMPUTED)        MODETR
C     OMMOD(N)   R*4    VAR   OUT (COMPUTED FOR N .GE. KST(NMODE))   MODETR
C     VPMOD(N)   R*4    VAR   OUT (COMPUTED FOR N .GE. KST(NMODE))   MODETR
C     NROW       I*4    NO    INP                                    MODETR
C     NCOL       I*4    NO    INP                                    MODETR
C     INMODE     I*4    VAR   INP                                    MODETR
C     OM         R*4    VAR   INP                                    MODETR
C     VP         R*4    VAR   INP                                    MODETR
C     KRUD       I*4    NO    OUT                                    MODETR
C                                                                    MODETR
C COMMON STORAGE USED                                                MODETR
C     COMMON IMAX,CI,VXI,VYI,HI,OMEGAC,VPHSEC,THETK                  MODETR
C                                                                    MODETR
C     IMAX       I*4    NO    INP                                    MODETR
C     CI         R*4    100   INP                                    MODETR
C     VXI        R*4    100   INP                                    MODETR
C     VYI        R*4    100   INP                                    MODETR
C     HI         R*4    100   INP                                    MODETR
C     OMEGAC     R*4    NO    OUT (USED INTERNALLY)                  MODETR
C     VPHSEC     R*4    NO    OUT (USED INTERNALLY)                  MODETR
C     THETK      R*4    NO    INP                                    MODETR
C                                                                    MODETR
C                   ----INPUTS----                                   MODETR
C                                                                    MODETR
C     IST        =ROW INDEX OF START POINT, WHICH MUST LIE BELOW LINE  MODETR
C                 OF DEMARCATION                                     MODETR
C     JST        =COLUMN INDEX OF START POINT                        MODETR
C     NMODE      =NUMBER LABELLING MODE TO BE TABULATED              MODETR
C     KST(NMODE) =INDEX OF OMMOD AND VPMOD CORRESPONDING TO FIRST    MODETR
C                 POINT TABULATED.                                   MODETR
C     NROW       =NUMBER OF ROWS IN INMODE ARRAY                     MODETR
```

```
C     NCOL         =NUMBER OF COLUMNS IN INMODE ARRAY
C     INMODE       =ARRAY WHOSE (J-1)*NROW+I-TH ELEMENT IS THE SIGN OF
C                   THE NORMAL MODE DISPERSION FUNCTION WHEN OMEGA=OM(J)
C                   VPHSE=VP(I).
C     OM           =VECTOR OF FREQUENCIES AT WHICH INMODE IS TABULATED.
C     VP           =VECTOR OF PHASE VELOCITIES AT WHICH INMODE IS
C                   TABULATED.
C     IMAX         =NUMBER OF ATMOSPHERIC LAYERS OF FINITE THICKNESS.
C     CI(I)        =SOUND SPEED IN I-TH LAYER IN KM/SEC.
C     VXI(I)       =X COMPONENT OF WIND VELOCITY IN I-TH LAYER IN KM/SEC
C     VYI(I)       =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER IN KM/SEC
C     HI(I)        =THICKNESS IN KM OF I-TH LAYER
C     THETK        =PHASE VELOCITY DIRECTION IN RADIANS W.R.T. X AXIS
C
C                   ----OUTPUTS----
C
C     KFIN(NMODE)   =INDEX OF OMMOD AND VPMOD CORRESPONDING TO LAST
C                    POINT TABULATED.
C     OMMOD(N)     =ANGULAR FREQUENCY OF POINTS ON DISPERSION CURVE.
C                   N=KST(NMODE) UP TO KFIN(NMODE) CORRESPONDS TO NMODE
C                   MODE.
C     VPMOD(N)     =PHASE VELOCITY OF POINTS ON DISPERSION CURVE.
C                   N=KST(NMODE) UP TO KFIN(NMODE) CORRESPONDS TO NMODE
C                   MODE.
C     KRUD          =FLAG INDICATING IF ANY POINTS ON DISPERION CURVE
C                    HAVE BEEN FOUND. 1 IF YES, -1 IF NO.
C     OMEGAC       =INTERNALLY USED FREQUENCY TRANSMITTED AMONG SUB-
C                   ROUTINES THROUGH COMMON
C     VPHSEC       =INTERNALLY USED PHASE VELOCITY TRANSMITTED AMONG
C                   SUBROUTINES THROUGH COMMON.
C
C                   ----EXAMPLE----
C
C     SUPPOSE THE TABLE OF INMODE VALUES IS AS SHOWN BELOW WITH
C
C         ++++++++++++++    NROW=7, NCOL=14
C         ++++++++++++++
C         ---+++++++++++    OM=.1,.2,.3,.4,.5,.6,.7,.8,.9,1.0,1.1,1.2,1.3
C         ---+++++++++++       1.4
C         ++------------
C         ++------------    VP=.5,.45,.40,.35,.30,.25,.20
C         +++-----------
C                           NMODE=2, IST=3, JST=1, KST(L)=7
C
C     THEN ONE MIGHT FIND KRUD=1, KFIN(2)=23, AND
C
C         OMMOD(7)=.1   VPMOD(7)=.43       OMMOD(16)=.75 VPMOD(16)=.3
C         OMMOD(8)=.2   VPMOD(8)=.42       OMMOD(17)=.8  VPMOD(17)=.29
C         OMMOD(9)=.3   VPMOD(9)=.41       OMMOD(18)=.9  VPMOD(18)=.285
C         OMMOD(10)=.33 VPMOD(10)=.4       OMMOD(19)=1.0 VPMOD(19)=.28
C         OMMOD(11)=.36 VPMOD(11)=.35      OMMOD(20)=1.1 VPMOD(20)=.27
C         OMMOD(12)=.40 VPMOD(12)=.34      OMMOD(21)=1.2 VPMOD(21)=.265
C         OMMOD(13)=.50 VPMOD(13)=.33      OMMOD(22)=1.3 VPMOD(22)=.26
C         OMMOD(14)=.60 VPMOD(14)=.32      OMMOD(23)=1.4 VPMOD(23)=.255
C         OMMOD(15)=.70 VPMOD(15)=.31
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),INMODE(1),OM(1),VP(1)
      COMMON IMAX,CI,VXI,VYI,HI,OMEGAC,VPHSEC,THETK
```

```
C
C FUNCTIONS FNMOD1 AND FNMOD2 ARE USED AS ARGUMENTS OF RTMI
      EXTERNAL FNMOD1,FNMOD2
C
C INDEX OF FIRST POINT ON DISPERSION CURVE IS LABELLED AS K
      K=KST(NMODE)
C
      J5=(JST-1)*NROW+IST
      I0=INMODE(J5)
C
C WE CHECK TO SEE IF POINT ABOVE (IST,JST) HAS A DIFFERENT INMODE
    2 IF(IST .EQ. 1) GO TO 5
      J6=(JST-1)*NROW+IST-1
      IUP=INMODE(J6)
      IF(IUP .EQ. -I0) GO TO 10
C
C IF IT DOESNT, WE CHECK THE POINT ON THE RIGHT.  WE CAN ALSO ARRIVE AT
C 5 FROM 2 IF IST=1.
    5 IF(JST .EQ. NCOL) GO TO 8
      J7=(JST)*NROW+IST
      ISID=INMODE(J7)
      IF(ISID .EQ. -I0) GO TO 15
C
C IF WE ARRIVE AT 8, WE CANNOT FIND A POINT EITHER ABOVE OR TO THE RIGH
C OF (IST,JST) WHICH HAS A INMODE OF OPPOSITE SIGN.
    8 KRUD=-1
      RETURN
C
C WE ASSIGN A TYPE INDEX TO THE POINT (IST,JST).  SEE DESCRIPTION OF
C NXTPNT FOR DEFINITION OF TYPE INDEX.
   10 ITYP1=1
C
C OPPOSITE SIGN ABOVE
      GO TO 20
C
   15 ITYP1=2
C OPPOSITE SIGN TO RIGHT
C
C WE NOW CAN IDENTIFY OUR FIRST BRACKETING
   20 I1=IST
      J1=JST
C
C STATEMENT 25 IS START OF LOOP TERMINATING AT 190.  EACH PASSAGE THROU
C LOOP GENERATES A NEW POINT ON THE DISPERSION CURVE.
   25 IF(ITYP1 .EQ. 2) GO TO 50
C
C CALCULATION IF ITYP1=1.  STORE FREQUENCY IN COMMON.  FIND PHASE VELO-
C CITY WITHIN BRACKETED INTERVAL.
      OMEGAC=OM(J1)
      VDOWN=VP(I1)
      VUP=VP(I1-1)
      EPS=1.E-6
      CALL RTMI(VA,F,FNMOD1,VDOWN,VUP,EPS,6,IER)
      OMMOD(K)=OMEGAC
      VPMOD(K)=VA
      GO TO 100
C
C CALCULATION IF ITYP1=2.  STORE PHASE VELOCITY IN COMMON.  FIND FREQUE
C IN BRACKETED INTERVAL.
   50 VPHSEC=VP(I1)
      OMLEF=OM(J1)
      OMRIT=OM(J1+1)
      EPS=(1.E-6)*OMRIT
      CALL RTMI(OMA,F,FNMOD2,OMLEF,OMRIT,EPS,6,IER)
```

```
      OMMOD(K)=OMA
      VPMOD(K)=VPHSEC
C
  180 CONTINUE
C WE HAVE NOW FOUND THE K-TH POINT.  WE DO NOT YET KNOW IF THIS IS THE
C FINAL POINT FOR THE NMODE-TH MODE.  HOWEVER, WE SET KFIN(NMODE)=K
      KFIN(NMODE)=K
C WHEN THE SUBROUTINE RETURNS, THE CURRENT STORED KFIN(NMODE) WILL BE
C THE CORRECT ONE.
C
C WE NOW PREPARE THE SEARCH FOR THE NEXT POINT.
      K=K+1
  179 CALL NXTPNT(I1,J1,ITYP1,I2,J2,ITYP2,NROW,NCOL,INMODE,KUDOS)
  190 IF( KUDOS .EQ. -1) GO TO 200
      I1=I2
      J1=J2
      ITYP1=ITYP2
  190 GO TO 25
C
  200 CONTINUE
C WE CONTINUE HERE AFTER AN UNSUCCESSFUL ATTEMPT TO FIND THE NEXT POINT
C PROVIDING WE HAVE FOUND AT LEAST ONE POINT, WE CAN EXIT WITH KRUD=1.
      IF( K .LE. KST(NMODE) ) GO TO 8
      KRUD=1
      RETURN
C
      END
```

```
      SUBROUTINE MODLST(MOFND,OMMOD,VPMOD,AKI,KST,KFIN)                  MODLST
C     MODLST (SUBROUTINE)                6/19/68   LAST CARD IN DECK IS MODLST
C                                                                        MODLST
C                                                                        MODLST
C                    ----ABSTRACT----                                    MODLST
C                                                                        MODLST
C                                                                        MODLST
C TITLE - MODLST                                                         MODLST
C         TABULATION OF SELECTED POINTS ON THE PHASE VELOCITY (VPHSE) VS MODLST
C         ANGULAR FREQUENCY (OMEGA) CURVES OF SELECTED MODES             MODLST
C                                                                        MODLST
C                NO COMPUTATION OR CHANGING OF UNITS IS PERFORMED BY SUB- MODLST
C                ROUTINE MODLST, IT MERELY PRINTS OUT THE INPUT IN LABELE MODLST
C                AND ORDERED FASHION.                                    MODLST
C                                                                        MODLST
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            MODLST
C                                                                        MODLST
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                MODLST
C                                                                        MODLST
C                                                                        MODLST
C                    ----USAGE----                                       MODLST
C                                                                        MODLST
C     NO SUBROUTINES ARE CALLED.                                         MODLST
C                                                                        MODLST
C     KFIN, OMMOD, VPMOD, KST WILL ASSUME THE DIMENSIONS SPECIFIED IN    MODLST
C     THE CALLING PROGRAM.  (DIMENSION OF KST AND KFIN MUST BE .GE. NMF  MODLST
C                                                                        MODLST
C FORTRAN USAGE                                                          MODLST
C                                                                        MODLST
C     CALL MODLST (MOFND,OMMOD,VPMOD,KST,KFIN)                           MODLST
C                                                                        MODLST
C INPUTS                                                                 MODLST
C                                                                        MODLST
C    MOFND      NUMBER OF MODES TO BE PRINTED OUT.                       MODLST
C      I*4                                                               MODLST
C                                                                        MODLST
C    OMMOD      VECTOR STORING ANGULAR FREQUENCY COORDINATE OF POINTS ON MODLST
C      R*4(D)   DISPERSION CURVES.  MODE M IS STORED FROM ELEMENT KST(M) MODLST
C               THROUGH ELEMENT KFIN(M).  ( RAD/SEC )                    MODLST
C                                                                        MODLST
C    VPMOD      VECTOR STORING PHASE VELOCITY COORDINATE OF POINTS ON    MODLST
C      R*4(D)   DISPERSION CURVES.  MODE M IS STORED FROM ELEMENT KST(M) MODLST
C               THROUGH ELEMENT KFIN(M).  ( KM/SEC )                     MODLST
C                                                                        MODLST
C    KST        SEE OMMOD AND VPMOD ABOVE.                               MODLST
C      I*4(D)                                                            MODLST
C                                                                        MODLST
C    KFIN       SEE OMMOD AND VPMOD ABOVE.                               MODLST
C      I*4(D)                                                            MODLST
C                                                                        MODLST
C OUTPUTS                                                                MODLST
C                                                                        MODLST
C    THE OUTPUT IS AN ORDERED AND LABELED PRINT OUT OF THE INPUTS, EX-   MODLST
C    CLUDING KST AND KFIN.  ( SEE EXAMPLE BELOW. )                       MODLST
C                                                                        MODLST
C                                                                        MODLST
C                    ----EXAMPLE----                                     MODLST
C                                                                        MODLST
C CALLING PROGRAM                                                        MODLST
C                                                                        MODLST
C     DIMENSION KST(2),KFIN(2),OMMOD(5),VPMOD(5)                         MODLST
C     MOFND = 2                                                          MODLST
C     KST = 1,3                                                          MODLST
C     KFIN = 2,5                                                         MODLST
```

```
C     OMMOD = 0.1,0.2,0.1,0.15,0.2
C     VPMOD = 1.0,2.0,2.0,2.5,3.0
C     CALL MODLST (MOFNO,OMMOD,VPMOD,KST,KFIN)
C
C PRINT OUT
C
C                    TABULATION OF FIRST     2 MODES
C
C
C
C                              MODE   1
C
C
C          OMEGA (RAD/SEC)          VPHSE (KM/SEC)
C
C                 0.100000                1.000000
C                 0.200000                2.000000
C
C
C
C                              MODE   2
C
C
C          OMEGA (RAD/SEC)          VPHSE (KM/SEC)
C
C                 0.100000                2.000000
C                 0.150000                2.500000
C                 0.200000                3.000000
C
C
C END OF EXAMPLE
C
C
C                ----PROGRAM FOLLOWS BELOW----
C
C
C
C VARIABLE DIMENSIONING
      DIMENSION KFIN(1),OMMOD(1),VPMOD(1),KST(1)
      DIMENSION AKI(1000)
      WRITE(6,11) MOFNO
   11 FORMAT(1H1,25X,19HTABULATION OF FIRST, I6,6H MODES)
      DO 100 II=1,MOFNO
      WRITE (6,21) II
   21 FORMAT(1H ///,1H ,35X, 5HMODE ,I3//, 1H ,12X,15HOMEGA (RAD/SEC),
     110X,14HVPHSE (KM/SEC),10X,15HAKI (NEPERS/KM)  /)
      K1=KST(II)
      K2 =KFIN(II)
      DO 100 J=K1,K2
      OMEGA=OMMOD(J)
      VPHSE=VPMOD(J)
      AKIPR=AKI(J)
   31 FORMAT (1H ,12X,F15.8,10X,F14.8,10X,5E12.5)
      WRITE (6,31) OMEGA,VPHSE,AKIPR
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MPOUT(OM1,OM2,V1,V2,NOM,NVP,INMODE,OM,V,THETK)          MPOUT
C     MPOUT (SUBROUTINE)                    7/19/68    LAST CARD IN DECK IS MPOUT
C                                                                        MPOUT
C                                                                        MPOUT
C                        ----ABSTRACT----                                MPOUT
C                                                                        MPOUT
C TITLE - MPOUT                                                          MPOUT
C          TABULATION OF NORMAL MODE DISPERSION FUNCTION SIGN AT POINTS  MPOUT
C          IN A RECTANGULAR REGION OF FREQUENCY - PHASE VELOCITY PLANE   MPOUT
C                                                                        MPOUT
C                  THE VECTOR V OF PHASE VELOCITIES IS CONSTRUCTED BY TAKIN MPOUT
C                  VALUES AT INTERVALS OF ((V2-V1)/(NVP-1)) FROM V2 DOWN TO MPOUT
C                  V1.  SIMILARLY, VECTOR OM OF ANGULAR FREQUENCIES IS CON- MPOUT
C                  STRUCTED BY TAKING VALUES AT INTERVALS OF ((OM2-OM1)/   MPOUT
C                  (NOM-1)) FROM OM1 UP TO OM2.  NEXT, MATRIX INMODE IS CON MPOUT
C                  STRUCTED WITH NVP ROWS AND NOM COLUMNS.  SINCE INMODE IS MPOUT
C                  STORED IN VECTOR FORM, COLUMN AFTER COLUMN, ELEMENT J IN MPOUT
C                  ROW I IS REPRESENTED AS INMODE((J-1)*NVP + I).  THE VALU MPOUT
C                  OF THIS ELEMENT IS DETERMINED BY CALLING SUBROUTINE NMOF MPOUT
C                  TO EVALUATE THE NORMAL MODE DISPERSION FUNCTION, FPP, FO MPOUT
C                  FREQUENCY OM(J) AND PHASE VELOCITY V(I).  IF FPP DOES NO MPOUT
C                  EXIST, THE ELEMENT IS SET EQUAL TO 5.  OTHERWISE THE ELE MPOUT
C                  MENT WILL BE 1 TIMES THE SIGN OF FPP.                MPOUT
C                                                                        MPOUT
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            MPOUT
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                MPOUT
C                                                                        MPOUT
C                                                                        MPOUT
C                        ----USAGE----                                   MPOUT
C     VARIABLES OM,V,INMODE MUST BE DIMENSIONED IN CALLING PROGRAM       MPOUT
C     FORTRAN SUBROUTINE NMOFN (DESCRIBED ELSEWHERE IN THIS SERIES) IS   MPOUT
C     CALLED                                                             MPOUT
C                                                                        MPOUT
C FORTRAN USAGE                                                          MPOUT
C     CALL MPOUT(OM1,OM2,V1,V2,NOM,NVP,INMODE,OM,V,THETK)                MPOUT
C                                                                        MPOUT
C INPUTS                                                                 MPOUT
C                                                                        MPOUT
C       OM1       MINIMUM ANGULAR FREQUENCY TO BE CONSIDERED (RADIANS / SE MPOUT
C         R*4                                                            MPOUT
C                                                                        MPOUT
C       OM2       MAXIMUM ANGULAR FREQUENCY TO BE CONSIDERED (RADIANS / SE MPOUT
C         R*4                                                            MPOUT
C                                                                        MPOUT
C       V1        MINIMUM PHASE VELOCITY TO BE CONSIDERED (KM / SEC)     MPOUT
C         R*4                                                            MPOUT
C                                                                        MPOUT
C       V2        MAXIMUM PHASE VELOCITY TO BE CONSIDERED (KM / SEC)     MPOUT
C         R*4                                                            MPOUT
C       NOM       NUMBER OF FREQUENCIES TO BE CONSIDERED (NO. OF ELEMENTS MPOUT
C         I*4     IN OM AND NO. OF COLUMNS IN INMODE)                    MPOUT
C                                                                        MPOUT
C       NVP       NUMBER OF PHASE VELOCITIES TO BE CONSIDERED (NO. OF ELE- MPOUT
C         I*4     MENTS IN V AND NO. OF ROWS IN INMODE)                  MPOUT
C                                                                        MPOUT
C       THETK     DIRECTION OF PHASE VELOCITY MEASURED COUNTER CLOCKWISE MPOUT
C         R*4     FROM X-AXIS (RADIANS)                                  MPOUT
C                                                                        MPOUT
C OUTPUTS                                                                MPOUT
C                                                                        MPOUT
C       INMODE    MATRIX OF NORMAL MODE DISPERSION FUNCTION SIGNS (SEE   MPOUT
C         I*4(D)  ABSTRACT ABOVE FOR EXPLANATION OF ELEMENT VALUES)      MPOUT
C                                                                        MPOUT
C       OM        VECTOR OF NOM VALUES OF ANGULAR FREQUENCY AT EQUAL INTER MPOUT
C         R*4(D)  VALS FROM OM1 TO OM2 INCLUSIVE (RADIANS / SEC).        MPOUT
C                                                                        MPOUT
C       V         VECTOR OF NVP VALUES OF PHASE VELOCITY AT EQUAL INTERVAL MPOUT
C         R*4(D)  FROM V2 TO V1 INCLUSIVE (KM / SEC)                     MPOUT
C                                                                        MPOUT
C                                                                        MPOUT
C                        ----EXAMPLE----                                 MPOUT
C                                                                        MPOUT
```

```
C CALLING PROGRAM
C    DIMENSION OM(3),V(3),INMOD(9)
C    OM1 = 1.0
C    OM2 = 3.0
C    V1 = 1.0
C    V2 = 3.0
C    NOM = 3
C    NVP = 3
C    THETK = 0.0
C    CALL MPOUT (OM1,OM2,V1,V2,NOM,NVP,INMODE,OM,V,THETK)
C    END
C
C UPON RETURN FROM MPOUT, OM AND V WILL HAVE THE FOLLOWING VALUES
C    OM = 1.0 , 2.0 , 3.0
C    V = 3.0 , 2.0 , 1.0
C EACH OF THE NINE ELEMENTS CF INMODE WILL BE -1, 1 OR 5 AS DETERMINED
C BY THE NORMAL MODE DISPERSION FUNCTION (SEE ABSTRACT ABOVE)
C
C
C                  ----PROGRAM FOLLOWS BELOW----
C
C
C
C VARIABLE DIMENSIONING
      DIMENSION OM(1),V(1),INMODE(1)
      COMMON IMAX,CI(100),VXI(100),VYI(100),HI(100)
C
C INTERVAL BETWEEN SUCCESSIVE ELEMENTS OF OM IS DETERMINED
      DELOM=(OM2-OM1)/(NOM-1)
C
C INTERVAL BETWEEN SUCCESSIVE ELEMENTS OF V IS DETERMINED
      DELV =(V2 - V1)/(NVP-1)
C
C VECTOR V IS CONSTRUCTED WITH V(I) DROPPING FROM V2TO V1 AS I GOES FRO
C 1 TO NVP
      V(1)=V2
      DO 10 I=2,NVP
   10 V(I)=V(I-1)-DELV
C
C OM(J) GOES FROM OM1 TO OM2 AS J GOES FROM 1 TO NOM
      DO 90 J=1,NOM
      OM(J) = OM1  +(J-1)*DELOM
C
C FOR A FIXED VALUE OF J, ALL VALUES OF I FROM 1 THROUGH NVP ARE CONSID
C ERED, THUS EVALUTING COLUMN J OF INMODE
      DO 90 I=1,NVP
C
C IJ IS NO. OF ELEMENT IN VECTOR REPRESENTATION OF INMODE WHICH CORRES-
C PONDS TO ELEMENT J OF ROW I IN MATRIX FORM OF INMODE
      IJ=(J-1)*NVP + I
      VPHSE=V(I)
C
C NMOFN IS CALLED TO EVAUATE THE NORMAL MODE DISPERSION FUNCTION FOR
C FREQUENCY OM(J) AND PHASE VELOCITY V(I)
      CALL NMOFN(OM(J),VPHSE,THETK,L,FPP,K)
C
C WHEN NORMAL MODE DISPERSION FUNCTION DOES NOT EXIST (L.EQ.-1), INMODE
C (IJ) = 5
      IF(L .EQ. -1) GO TO 50
C
C WHEN THE FUNCTION DOES EXIST AND IS FPP, INMODE(IJ) = 1*FPP/ABS(FPP)
      INMODE(IJ) = 1
      IF (FPP.LE.0.0) INMODE(IJ) = -1
      GO TO 80
   50 INMODE(IJ)=5
   80 CONTINUE
   90 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE NAMPDE(ZSCRCE,ZOBS,OMEGA,VPHSE,AKI,THETK,AMPLTD,NPRNT) NAMPDE
C     NAMPDE (SUBROUTINE)                6/27/69   LAST CARD IN DECK IS NAMPDE
C                                                                      NAMPDE
C                                                                      NAMPDE
C                  ----ABSTRACT----                                    NAMPDE
C                                                                      NAMPDE
C TITLE - NAMPDE                                                       NAMPDE
C         PROGRAM TO DETERMINE AN AMPLITUDE FACTOR AMPLTD OF A GUIDED  NAMPDE
C         MODE EXCITED BY A POINT ENERGY SOURCE IN THE ATMOSPHERE.  THE NAMPDE
C         SOURCE IS AT ALTITUDE ZSCRCE KM AND THE OBSERVER IS AT ALTITUD NAMPDE
C         ZOBS IN KM.  THE PARTICULAR AMPLTD COMPUTED CORRESPONDS TO AN NAMPDE
C         ANGULAR FREQUENCY OMEGA (RAD/SEC), A PHASE VELOCITY VPHSE    NAMPDE
C         (KM/SEC), AND A PHASE VELOCITY DIRECTION THETK (RADIANS) REC- NAMPDE
C         KONED COUNTER-CLOCKWISE FROM THE X AXIS.  PARAMETERS DEFINING NAMPDE
C         THE AMBIENT ATMOSPHERE ARE PRESUMED TO BE STORED IN COMMON.  NAMPDE
C         THE NORMAL MODE DISPERSION FUNCTION NMDF IS PRESUMED TO VANISH.NAMPDE
C         FOR ARGUMENTS OMEGA,VPHSE,THETK.                             NAMPDE
C                                                                      NAMPDE
C         THE ACTUAL DEFINITION OF AMPLTD IS AS FOLLOWS.  LET S1(Z) AND NAMPDE
C         S2(Z) BE THE SOLUTIONS OF THE RESIDUAL EQUATIONS             NAMPDE
C                                                                      NAMPDE
C             D(S1)/DZ = (A11)*S1 + (A12)*S2                    (1-A) NAMPDE
C             D(S2)/DZ = (A21)*S1 + (A22)*S2                    (1-B) NAMPDE
C                                                                      NAMPDE
C         WHERE THE MATRIX A IS AS COMPUTED BY AAAA AND AS DEFINED BY  NAMPDE
C         A.D.PIERCE, J. COMP. PHYS., VOL. 1, NO. 3, FEB.,1967, PP. 343- NAMPDE
C         366, EQS. 11.  THE ELEMENTS OF A SHOULD BE CONSIDERED AS FUNC- NAMPDE
C         TIONS OF ALTITUDE.  WE DEFINE THE REDUCED PRESSURE ZFN(Z) AS NAMPDE
C                                                                      NAMPDE
C             ZFN(Z)= (G/C)*S1 - C*S2                             (2) NAMPDE
C                                                                      NAMPDE
C         WHERE G IS ACCELERATION OF GRAVITY AND C IS SOUND SPEED.  THEN NAMPDE
C                                                                      NAMPDE
C                                                                      NAMPDE
C                            S2(ZSCRCE)*ZFN(ZOBS)                      NAMPDE
C           AMPLTD = (1/2)* ----------------------                 (3) NAMPDE
C                           BOM(ZSCRCE)*INTEGRAL                       NAMPDE
C                                                                      NAMPDE
C         WHERE                                                        NAMPDE
C                                                                      NAMPDE
C             BOM(Z)=OMEGA -KX*VX(Z) -KY*VY(Z) .                  (4) NAMPDE
C                                                                      NAMPDE
C         IS THE DOPPLER SHIFTED ANGULAR FREQUENCY.  THE INTEGRAL IS 1/2 NAMPDE
C         OF THE I-SUB1 DEFINED BY A.D.PIERCE, J. ACOUST. SOC. AMER.,  NAMPDE
C         VOL. 37, NO. 2, FEB.,1965, PP. 218-227, EQ. (51).  SPECIFICALL NAMPDE
C                                                                      NAMPDE
C         INTEGRAL = (INTEGRAL OVER Z FROM 0 TO INFINITY) OF           NAMPDE
C                                                                      NAMPDE
C                    (BOM*((KX*VX+KY*VY)/K)*YFN(Z)**2                 NAMPDE
C                    +(K*OMEGA/BOM**3)*ZFN(Z)**2                  (5) NAMPDE
C                                                                      NAMPDE
C         WHERE K IS THE MAGNITUDE OF THE WAVE-NUMBER VECTOR (KX,KY) AND NAMPDE
C                                                                      NAMPDE
C             YFN(Z) = (1/C)*S1(Z)                                (6) NAMPDE
C                                                                      NAMPDE
C PROGRAM NOTES                                                        NAMPDE
C                                                                      NAMPDE
C              THE INTEGRAL IS COMPUTED BY SUBROUTINE TOTINT IN TWO PAR NAMPDE
C              AS X3+X7, THE FIRST IS OBTAINED BY CALLING TOTINT WITH  NAMPDE
C              IT=3, WHILE THE SECOND IS OBTAINED BY CALLING TOTINT WIT NAMPDE
```

```
C               IT=7.  THE IT PARAMETER GOVERNS THE CHOICE OF COEFFICIEN NAMPOE
C               A1, A2, A3 RETURNED TO TOTINT BY SUBROUTINE USEAS.  FOR  NAMPOE
C               FURTHER INFORMATION, SEE THE DOCUMENTATION ON TOTINT AND NAMPOE
C               USEAS.                                                   NAMPOE
C                                                                        NAMPOE
C               THE NORMALIZATION OF S1 AND S2 CANNOT AFFECT AMPLTO.     NAMPOE
C               HOWEVER, TOTINT ACOOPTS NORMALIZATION WHERE              NAMPOE
C                     S1 =-SQRT(GG)*A12                                  NAMPOE
C                     S2 =SQRT(GG)*(GG+A11)                              NAMPOE
C               AT THE BOTTOM OF THE UPPER HALF SPACE.  THE NUMERATOR OF NAMPOE
C               EQ.(3) IS ACCORDINGLY COMPUTED WITH SAME NORMALIZATION.  NAMPOE
C               HERE GG=SQRT(A11**2+A12*A21).                            NAMPOE
C                                                                        NAMPOE
C               THE ONLY BOUNDARY CONDITION EXPLICITLY USED IS THE UPPER NAMPOE
C               BOUNDARY CONDITION WHEREBY BOTH S1(Z) AND S2(Z) DECREASE NAMPOE
C               EXPONENTIALLY WITH INCREASING HEIGHT IN THE UPPER        NAMPOE
C               HALFSPACE.  IF THIS CANNOT BE SATISFIED, THE PROGRAM     NAMPOE
C               RETURNS AMPLTO=0.  THIS WOULD IMPLY THAT THE POINT       NAMPOE
C               CONSIDERED IS PRACTICALLY IDENTICAL TO ONE WHERE OMEGA   NAMPOE
C               IS THE CUTOFF FREQUENCY FOR THE GUIDED MODE UNDER        NAMPOE
C               CONSIDERATION.                                           NAMPOE
C                                                                        NAMPOE
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             NAMPOE
C                                                                        NAMPOE
C AUTHOR     - A.D.PIERCE, M.I.T., JUNE,1968                             NAMPOE
C                                                                        NAMPOE
C                   ----CALLING SEQUENCE----                             NAMPOE
C                                                                        NAMPOE
C SEE SUBROUTINE PAMPOE                                                  NAMPOE
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                        NAMPOE
C     COMMON IMAX,CI,VXI,VYI,HI        (THESE MUST BE IN COMMON)         NAMPOE
C     CALL NAMPOE(ZSCRCE,ZOBS,OMEGA,VPHSE,THETK,AMPLTO,NPRNT)            NAMPOE
C                                                                        NAMPOE
C                   ----EXTERNAL SUBROUTINES REQUIRED----                NAMPOE
C                                                                        NAMPOE
C     TOTINT,MMMM,AAAA,USEAS,UPINT,ELINT,BBBB,CAI,SAI                    NAMPOE
C                                                                        NAMPOE
C     (THE FIRST THREE ARE EXPLICITLY CALLED.  THE REMAINING SUBROUTINE  NAMPOE
C      ARE IMPICITLY CALLED WHEN TOTINT IS CALLED.)                      NAMPOE
C                                                                        NAMPOE
C               ----ARGUMENT LIST----                                    NAMPOE
C                                                                        NAMPOE
C     ZSCRCE        R*4     NO      INP                                  NAMPOE
C     ZOBS          R*4     NO      INP                                  NAMPOE
C     OMEGA         R*4     NO      INP                                  NAMPOE
C     VPHSE         R*4     NO      INP                                  NAMPOE
C     THETK         R*4     NO      INP                                  NAMPOE
C     AMPLTO        R*4     NO      OUT                                  NAMPOE
C     NPRNT         I*4     NO      INP                                  NAMPOE
C                                                                        NAMPOE
C COMMON STORAGE USED                                                    NAMPOE
C     COMMON IMAX,CI,VXI,VYI,HI                                          NAMPOE
C                                                                        NAMPOE
C     IMAX          I*4     NO      INP                                  NAMPOE
C     CI            R*4     100     INP                                  NAMPOE
C     VXI           R*4     100     INP                                  NAMPOE
C     VYI           R*4     100     INP                                  NAMPOE
C     HI            R*4     100     INP                                  NAMPOE
C                                                                        NAMPOE
C                   ----INPUTS----                                       NAMPOE
C                                                                        NAMPOE
C     ZSCRCE        =HEIGHT OF SOURCE IN KM                              NAMPOE
C     ZOBS          =HEIGHT OF OBSERVER                                  NAMPOE
C     OMEGA         =ANGULAR FREQUENCY IN RADIANS/SEC                    NAMPOE
```

```
C     VPHSE         =PHASE VELOCITY IN KM/SEC
C     THETK         =PHASE VELOCITY DIRECTION (RADIANS) RECKONED
C     NPRINT        =PRINT OPTION INDICATOR (SEE NAM1 IN MAIN PROGRAM).
C                    COUNTER-CLOCKWISE FROM X AXIS.
C     IMAX          =NUMBER OF ATMOSPHERIC LAYERS WITH FINITE THICKNESS
C     CI(I)         =SOUND SPEED (KM/SEC) IN I-TH LAYER
C     VXI(I)        =X COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C     VYI(I)        =Y COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C     HI(I)         =THICKNESS IN KM OF I-TH LAYER
C
C                    ----OUTPUTS----
C
C     AMPLTO        =AMPLITUDE FACTOR FOR GUIDED WAVE EXCITED BY POINT
C                    ENERGY SOURCE.  UNITS ARE KM**(-1).
C
C                    ----EXAMPLE----
C
C     SUPPOSE THE ATMOSPHERE IS ISOTHERMAL AND THERE ARE NO WINDS.  THE
C     THERE IS ONLY ONE MODE, FOR WHICH VPHSE=C.  FURTHERMORE, YFN(Z)
C     AND S1(Z) ARE BOTH ZERO.  THE ZFN(Z) DECREASES EXPONENTIALLY
C     WITH HEIGHT AS EXP(-(0.3*G/C**2)).  THE RESULTING AMPLTO
C     SHOULD BE
C       AMPLTO=-(.3*G/C**2)*EXP(-.3*(G/C**2)*(ZOBS+ZSCRCE))
C     REGARDLESS OF VALUES OF OMEGA AND THETK.  IF C=1/3 KM/SEC,
C     G=.01 KM/SEC**2, ZOBS=0, ZSCRCE=0, THEN AMPLTO= .027 KM**(-1).
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      DIMENSION A(2,2),EM(2,2)
      DIMENSION ZIJZ(2),S1(2),S2(2),VXIJZ(2),VYIJZ(2),CIJZ(2)
C DIMENSION STATEMENTS ADDED IN THE DEBUG PROCESS
      DIMENSION LAYJZ(2),DELT(2),RPP(2,2),EMP(100,2,2),DUMMY(2,2)
      DIMENSION PHI1(100),PHI2(100)
      COMMON IMAX,CI,VXI,VYI,HI
      SAVE = CI(IMAX)
      IF(AKI .GE. 1.E-10) RETURN
C
C COMPUTE WAVE NUMBER VECTOR COMPONENTS
    1 CONTINUE
      AKX=(OMEGA/VPHSE)*COS(THETK)
      AKY=(OMEGA/VPHSE)*SIN(THETK)
C
C THE SOURCE AND OBSERVER LOCATIONS ARE NUMBERED ACCORDING TO HEIGHT
      IF(ZSCRCE .GT. ZOBS) GO TO 10
      ZIJZ(1)=ZSCRCE
      ZIJZ(2)=ZOBS
      NSCRCE=1
      NOBS=2
      GO TO 20
   10 ZIJZ(1)=ZOBS
      ZIJZ(2)=ZSCRCE
      NOBS=1
      NSCRCE=2
C
C WE DENOTE S1 AND S2 AT BOTTOM OF UPPER HALFSPACE BY F1 AND F2.  THEIR
C COMPUTATION IS AS FOLLOWS.
   20 IZMAX=2
      J=IMAX+1
      C=CI(J)
      VX=VXI(J)
```

```
      VY=VYI(J)
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)
      X=A(1,1)**2+A(1,2)*A(2,1)
      IF(X .LE. 0.0) GO TO 200
      G=SQRT(X)
      GRT=SQRT(G)
      F1=-GRT*A(1,2)
      F2=GRT*(A(1,1)+G)
C
C WE COMPUTE ZM REPRESENTING THE BOTTOM OF THE UPPER HALFSPACE
      ZM=0.0
      IF(IMAX .EQ. 0) GO TO 31
      DO 30 IC=1,IMAX
   30 ZM=ZM+HI(IC)
C
C WE STORE F1P,F2P,ZMP
   31 F1P=F1
      F2P=F2
      ZMP=ZM
C
C COMPUTATION OF LAYJZ(JZ) AND DELT(JZ)
C LAYJZ(JZ) IS THE INDEX OF THE LAYER IN WHICH ZIJZ(JZ) LIES,
C WHILE DELT(JZ) IS THE DISTANCE OF ZIJZ(JZ) ABOVE THE BOTTOM EDGE OF
C THE LAYER
      DO 35 JZ=1,2
      LAYJZ(JZ)=IMAX+1
   32 DELT(JZ)=ZIJZ(JZ)-ZM
      IF(DELT(JZ) .GT. 0.0) GO TO 35
      IF(LAYJZ(JZ) .EQ. 1) GO TO 35
      LAYJZ(JZ)=LAYJZ(JZ)-1
      J8=LAYJZ(JZ)
      ZM=ZM-HI(J8)
C AT THIS POINT ZM DENOTES THE BOTTOM OF THE LAYJZ(JZ) LAYER
      GO TO 32
   35 ZM=ZMP
C
C COMPUTATION OF EM MATRICES FOR ALL IMAX LAYERS OF FINITE THICKNESS
C EM(IP,JP) FOR I-TH LAYER IS STORED AS EMP(I,IP,JP)
      DO 36 I=1,IMAX
      C=CI(I)
      VX=VXI(I)
      VY=VYI(I)
      H=HI(I)
      CALL MMMM(OMEGA,AKX,AKY,C,VX,VY,H,EM)
      DO 36 IP=1,2
      DO 36 JP=1,2
   36 EMP(I,IP,JP)=EM(IP,JP)
C
C COMPUTATION OF RPP MATRIX.  THIS ACCOMPLISHES THE SAME AS CALLING
C SUBROUTINE RRRR
      RPP(1,1)=1.0
      RPP(1,2)=0.0
      RPP(2,1)=0.0
      RPP(2,2)=1.0
      DO 38 I=1,IMAX
      JASA=IMAX+1-I
      DO 37 IP=1,2
      DO 37 JP=1,2
   37 DUMMY(IP,JP)=EMP(JASA,IP,1)*RPP(1,JP)+EMP(JASA,IP,2)*RPP(2,JP)
      DO 38 IP=1,2
      DO 38 JP=1,2
   38 RPP(IP,JP)=DUMMY(IP,JP)
C
      QUOT = ABS(RPP(1,1))/(ABS(RPP(1,1))+ABS(RPP(1,2))+ABS(RPP(2,1))
```

```
     1  +ABS(RPP(2,2)))
      IF ( QUOT .LT. 0.1 ) GO TO 120
      F2BOT=F2P/RPP(1,1)
      GO TO 150
  120 QUOT = ABS(RPP(1,2))/(ABS(RPP(1,1))+ABS(RPP(1,2))+ABS(RPP(2,1))
     1  +ABS(RPP(2,2)))
      IF ( QUOT .LT. 0.1 ) GO TO 130
      F2BOT=-F1P/RPP(1,2)
      GO TO 150
  130 F2BOT=RPP(2,1)*F1P+RPP(2,2)*F2P
  150 F2BOT=F2BOT
      PHI1(1)=0.0
      PHI2(1)=F2BOT
      KTOUP=1
      K=IMAX+1
      PHI1(K)=F1P
      PHI2(K)=F2P
  331 T1=PHI1(K)
      T2=PHI2(K)
      K=K-1
      IF(K .EQ. 1) GO TO 400
      C=CI(K)
      VX=VXI(K)
      VY=VYI(K)
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)
      X=A(1,1)**2+A(1,2)*A(2,1)
      IF(X .GT. 0.0) GO TO 340
  333 PHI1(K)=EMP(K,1,1)*T1+EMP(K,1,2)*T2
      PHI2(K)=EMP(K,2,1)*T1+EMP(K,2,2)*T2
      GO TO 331
  340 D1=A(1,1)*T1+A(1,2)*T2
      D2=A(2,1)*T1+A(2,2)*T2
      IF ( D1 .LT. 0.0 .AND. T1 .LT. 0.0) GO TO 341
      IF ( D1 .GT. 0.0 .AND. T1 .GT. 0.0 ) GO TO 341
      IF ( D2 .LT. 0.0 .AND. T2 .LT. 0.0 ) GO TO 341
      IF ( D2 .GT. 0.0 .AND. T2 .GT. 0.0 ) GO TO 341
      GO TO 333
  341 CONTINUE
C AT THIS POINT THE CURRENT VALUE OF K IS NOT ZERO OR ONE
      KTOUP=K
      DO 360 K=2,KTOUP
      JET=K-1
      T1=PHI1(JET)
      T2=PHI2(JET)
      PHI1(K)=EMP(JET,2,2)*T1-EMP(JET,1,2)*T2
  360 PHI2(K)=-EMP(JET,2,1)*T1+EMP(JET,1,1)*T2
  400 NZC1 = 0
      NZC2 = 0
      IAP1MX = 1
      IAP2MX = 1
      AP1MX = ABS(PHI1(1))
      AP2MX = ABS(PHI2(1))
      DO 407 LNM1=1,IMAX
      LN = LNM1 + 1
      AP1P = ABS(PHI1(LN))
      IF (AP1P.LE.AP1MX) GO TO 403
      IAP1MX = LN
      AP1MX = AP1P
  403 AP2P = ABS(PHI2(LN))
      IF (AP2P.LE.AP2MX) GO TO 405
      AP2MX = AP2P
      IAP2MX = LN
  405 IF ((PHI1(LNM1)*PHI1(LN)).LT.0.0) NZC1 = NZC1 + 1
      IF ((PHI2(LNM1)*PHI2(LN)).LT.0.0) NZC2 = NZC2 + 1
```

```
  407 CONTINUE
      R1 = PHI1(IAP1MX)/AP2MX
      R2 = PHI2(IAP2MX)/AP2MX
      R3 = PHI2(1)/AP2MX
      WRITE (6,409) OMEGA,VPHSE,IAP1MX,R1,NZC1,IAP2MX,R2,NZC2,R3
  409 FORMAT (1H ,2F12.5,9X,I3,F12.5,9X,I3,9X,I3,F12.5,9X,I3,F12.5)
  415 DO 450 JZ=1,2
      IDA=LAYJZ(JZ)
      C=CI(IDA)
      VX=VXI(IDA)
      VY=VYI(IDA)
      CIJZ(JZ)=CI(IDA)
      VXIJZ(JZ)=VXI(IDA)
      VYIJZ(JZ)=VYI(IDA)
      IF(IDA .EQ. IMAX+1) GO TO 420
      IF(IDA .LE. KTOUP) GO TO 430
      JET=IDA+1
      H=HI(IDA)-DELT(JZ)
      CALL MMMM(OMEGA,AKX,AKY,C,VX,VY,H,EM)
      S1(JZ)=EM(1,1)*PHI1(JET)+EM(1,2)*PHI2(JET)
      S2(JZ)=EM(2,1)*PHI1(JET)+EM(2,2)*PHI2(JET)
      GO TO 450
  420 EON=EXP(-G*DELT(JZ))
      S1(JZ)=F1P*EON
      S2(JZ)=F2P*EON
      GO TO 450
  430 H=DELT(JZ)
      CALL MMMM(OMEGA,AKX,AKY,C,VX,VY,H,EM)
      S1(JZ)=EM(2,2)*PHI1(IDA)-EM(1,2)*PHI2(IDA)
      S2(JZ)=-EM(2,1)*PHI1(IDA)+EM(1,1)*PHI2(IDA)
  450 CONTINUE
C
C AT THIS POINT S1(JZ),S2(JZ),CIJZ(JZ), ETC. ARE STORED FOR JZ=1 AND 2.
C WE COMPUTE THE DOPPLER SHIFTED ANGULAR FREQUENCY AT SOURCE ALTITUDE.
  100 BOM=OMEGA-AKX*VXIJZ(NSCRCE)-AKY*VYIJZ(NSCRCE)
C
C WE COMPUTE ZFN AT OBSERVER ALTITUDE
      ZFN=(.0098/CIJZ(NOBS))*S1(NOBS)-CIJZ(NOBS)*S2(NOBS)
C
C HERE WE TAKE THE ACCELERATION OF GRAVITY TO BE .0098 KM/SEC**2.
C
C COMPUTATION OF INTEGRALS
      IT=3
      CALL TOTINT(OMEGA,AKX,AKY,IT,L,X3,PHI1,PHI2)
      IF(L .EQ. -1) GO TO 200
      IT=7
      CALL TOTINT(OMEGA,AKX,AKY,IT,L,X7,PHI1,PHI2)
      IF(L .EQ. -1) GO TO 200
C
C FINAL ANSWER
      AMPLTD= 0.5*S2(NSCRCE)*ZFN/((X3+X7)*BOM)
      CI(IMAX) = SAVE
      RETURN
C
C IF YOU ARRIVE HERE, THE UPPER BOUNDARY CONDITION COULD NOT BE SATISFI
  200 AMPLTD=0.0
      CI(IMAX) = 1.E5
      GO TO 1
C
      END
```

```
      SUBROUTINE NMOFN(OMEGA,VPHSE,THETK,L,FPP,K)                      NMOFN
C     NMOFN (SUBROUTINE)                 7/25/68   LAST CARD IN DECK IS NMOFN
C                                                                      NMOFN
C                    ----ABSTRACT----                                  NMOFN
C                                                                      NMOFN
C TITLE - NMOFN                                                        NMOFN
C          SUBROUTINE TO COMPUTE THE NORMAL MODE DISPERSION FUNCTION FPP NMOFN
C          FOR GIVEN ANGULAR FREQUENCY OMEGA, PHASE VELOCITY MAGNITUDE  NMOFN
C          VPHSE AND PHASE VELOCITY DIRECTION THETK.  FPP SHOULD VANISH NMOFN
C          IF BOTH UPPER AND LOWER BOUNDARY CONDITIONS ARE SATISFIED FOR NMOFN
C          THE SOLUTIONS OF THE RESIDUAL EQUATIONS                      NMOFN
C                                                                      NMOFN
C              D(PHI1)/DZ = A(1,1)*PHI1(Z) + A(1,2)*PHI2(Z)             NMOFN
C                                                                      NMOFN
C              D(PHI2)/DZ = A(2,1)*PHI1(Z) + A(2,2)*PHI2(Z)             NMOFN
C                                                                      NMOFN
C          WHERE THE ELEMENTS OF THE MATRIX A VARY WITH HEIGHT Z, BUT ARE NMOFN
C          CONSTANT IN EACH LAYER OF A MULTILAYER ATMOSPHERE.  THE ELEMEN NMOFN
C          OF A ARE FUNCTIONS OF OMEGA, AKX AND AKY AS DESCRIBED IN     NMOFN
C          SUBROUTINE AAAA WHERE                                        NMOFN
C                                                                      NMOFN
C              AKX=OMEGA*COS(THETK)/VPHSE                               NMOFN
C                                                                      NMOFN
C              AKY=OMEGA*SIN(THETK)/VPHSE                               NMOFN
C                                                                      NMOFN
C          THE FUNCTION FPP IS DEFINED AS THE VALUE OF PHI1 AT THE GROUND NMOFN
C          (Z=0) WHEN (1) THE UPPER BOUNDARY CONDITION OF PHI1 AND PHI2 NMOFN
C          DECREASING EXPONENTIALLY WITH HEIGHT IN THE UPPER HALFSPACE  NMOFN
C          IS SATISFIED, AND (2) PHI1 AND PHI2 AT THE BOTTOM OF THE UPPER NMOFN
C          HALFSPACE ARE GIVEN BY A(1,2) AND -(G+A(1,1)) WHERE          NMOFN
C          G=SQRT(A(1,1)**2+A(1,2)*A(2,1)).  THE ELEMENTS OF A HERE ARE NMOFN
C          THOSE APPROPRIATE TO THE UPPER HALFSPACE.  CONDITIONS (1) AND NMOFN
C          (2) ARE NOT INDEPENDENT.  CONDITION (1) IMPLIES THAT G**2 .GT. NMOFN
C          AND CONDITION (2) WITH G**2 POSITIVE IMPLIES (1).  IF G**2 IS NMOFN
C          NEGATIVE, FPP DOES NOT EXIST AND L=-1 IS RETURNED.  OTHERWISE NMOFN
C          L=1 IS RETURNED.                                             NMOFN
C                                                                      NMOFN
C PROGRAM NOTES                                                        NMOFN
C                                                                      NMOFN
C                THE PARAMETERS DEFINING THE MULTILAYER MODEL ATMOSPHERE NMOFN
C                ARE PRESUMED TO BE STORED IN COMMON.                   NMOFN
C                                                                      NMOFN
C                THE SUBROUTINE RRRR IS USED TO GENERATE THE MATRIX RPP NMOFN
C                WHICH CONNECTS SOLUTIONS OF THE RESIDUAL EQUATIONS AT  NMOFN
C                THE BOTTOM OF THE UPPER HALFSPACE TO SOLUTIONS AT THE  NMOFN
C                GROUND.  IN TERMS OF THIS MATRIX, THE NMDF IS GIVEN BY NMOFN
C                                                                      NMOFN
C                 FPP= RPP(1,1)*A(1,2)-RPP(1,2)*(G+A(1,1))              NMOFN
C                                                                      NMOFN
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)           NMOFN
C                                                                      NMOFN
C AUTHOR     - A.D.PIERCE, M.I.T., AUGUST,1968                         NMOFN
C                                                                      NMOFN
C                    ----CALLING SEQUENCE----                          NMOFN
C                                                                      NMOFN
C SEE SUBROUTINES LNGTHN,WIDEN,MPOUT                                   NMOFN
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                      NMOFN
C     COMMON IMAX,CI,VXI,VYI,HI  (THESE MUST BE STORED IN COMMON)      NMOFN
C     CALL NMOFN(OMEGA,VPHSE,THETK,L,FPP,K)                            NMOFN
C                                                                      NMOFN
C                    ----EXTERNAL SUBROUTINES REQUIRED----             NMOFN
C                                                                      NMOFN
C     RRRR,MMMM,AAAA,CAI,SAI                                           NMOFN
C                                                                      NMOFN
```

```
C                    ----ARGUMENT LIST----                                      NMOFN
C                                                                               NMOFN
C     OMEGA      R*4    NO    INP                                               NMOFN
C     VPHSE      R*4    NO    INP                                               NMOFN
C     THETK      R*4    NO    INP                                               NMOFN
C     L          I*4    NO    OUT                                               NMOFN
C     FPP        R*4    NO    OUT                                               NMOFN
C     K          I*4    NO    OUT  (ALWAYS RETURNED AS K=0)                     NMOFN
C                                                                               NMOFN
C COMMON STORAGE USED                                                           NMOFN
C     COMMON IMAX,CI,VXI,VYI,HI                                                 NMOFN
C                                                                               NMOFN
C     IMAX       I*4    NO    INP                                               NMOFN
C     CI         R*4    100   INP                                               NMOFN
C     VXI        R*4    100   INP                                               NMOFN
C     VYI        R*4    100   INP                                               NMOFN
C     HI         R*4    100   INP                                               NMOFN
C                                                                               NMOFN
C                    ----INPUTS----                                             NMOFN
C                                                                               NMOFN
C     OMEGA      =ANGULAR FREQUENCY IN RAD/SEC                                  NMOFN
C     VPHSE      =PHASE VELOCITY MAGNITUDE IN KM/SEC                            NMOFN
C     THETK      =PHASE VELOCITY DIRECTION RECKONED COUNTER CLOCKWISE           NMOFN
C                 FROM THE X AXIS IN RADIANS                                    NMOFN
C     IMAX       =NUMBER OF LAYERS OF FINITE THICKNESS                          NMOFN
C     CI(I)      =SOUND SPEED IN KM/SEC IN I-TH LAYER                           NMOFN
C     VXI(I)     =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)           NMOFN
C     VYI(I)     =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)           NMOFN
C     HI(I)      =THICKNESS IN KM OF I-TH LAYER OF I-TH THICKNESS               NMOFN
C                                                                               NMOFN
C                    ----OUTPUTS----                                            NMOFN
C                                                                               NMOFN
C     L          =1 IF NORMAL MODE DISPERSION FUNCTION EXISTS, -1 IF            NMOFN
C                 IT DOES NOT.                                                  NMOFN
C     FPP        =NORMAL MODE DISPERSION FUNCTION                               NMOFN
C     K          =DUMMY PARAMETER ALWAYS RETURNED AS K=0                        NMOFN
C                                                                               NMOFN
C                    ----PROGRAM FOLLOWS BELOW----                              NMOFN
C                                                                               NMOFN
C                                                                               NMOFN
C DIMENSION AND COMMON STATEMENTS LOCATING PARAMETERS DEFINING MODEL            NMOFN
C MULTILAYER ATMOSPHERE                                                         NMOFN
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)                               NMOFN
      COMMON IMAX,CI,VXI,VYI,HI                                                 NMOFN
C                                                                               NMOFN
      DIMENSION A(2,2),RPP(2,2)                                                 NMOFN
C                                                                               NMOFN
C COMPUTATION OF AKX AND AKY                                                    NMOFN
      AKX=OMEGA*COS(THETK)/VPHSE                                                NMOFN
      AKY=OMEGA*SIN(THETK)/VPHSE                                                NMOFN
C                                                                               NMOFN
C COMPUTATION OF MATRIX A AND G**2 FOR UPPER HALFSPACE                          NMOFN
      J=IMAX+1                                                                  NMOFN
      C=CI(J)                                                                   NMOFN
      VX=VXI(J)                                                                 NMOFN
      VY=VYI(J)                                                                 NMOFN
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)                                        NMOFN
      GUSQ=A(1,1)**2+A(1,2)*A(2,1)                                              NMOFN
C                                                                               NMOFN
      IF(GUSQ .GT. 0.0) GO TO 11                                                NMOFN
C                                                                               NMOFN
C GUSQ IS LESS THAN ZERO                                                        NMOFN
      L=-1                                                                      NMOFN
```

```
      RETURN
C
C GUSQ IS GREATER THAN ZERO
   11 L=1
      GU=SQRT(GUSQ)
C
C COMPUTATION OF RPP MATRIX
      CALL RPPP(OMEGA,AKX,AKY,RPP,K)
C
C COMPUTATION OF FPP
      FPP = RPP(1,1)*A(1,2) - RPP(1,2)*(GU+A(1,1))
C
      RETURN
      END
```

```
      SUBROUTINE NXMODE(IST,JST,NOM,NVP,INMODE,IFND,JFND,K)             NXMODE
      DIMENSION INMODE(1)                                               NXMODE
C     NXMODE (SUBROUTINE)                6/24/68   LAST CARD IN DECK IS NXMODE
C                                                                       NXMODE
C                                                                       NXMODE
C                    ----ABSTRACT----                                   NXMODE
C                                                                       NXMODE
C TITLE - NXMODE                                                        NXMODE
C          PROGRAM TO FIND A POINT WITH COORDINATES I=IFND,J=JFND IN AN  NXMODE
C          ARRAY WITH NOM COLUMNS AND NVP ROWS.  FOUND POINT CORRESPONDS NXMODE
C          TO STARTING POSITION FOR CALCULATION OF PHASE VELOCITY VERSUS NXMODE
C          FREQUENCY OF A PARTICULAR GUIDED MODE.  A TABLE OF VALUES OF  NXMODE
C          THE SIGN OF THE NORMAL MODE DISPERSION FUNCTION IS PRESUMED   NXMODE
C          TO BE STORED AS INMODE((J-1)*NVP+I) FOR EACH POINT (I,J) IN TH NXMODE
C          ARRAY.  DIFFERENT COLUMNS (J) CORRESPOND TO DIFFERENT FREQUEN- NXMODE
C          CIES WHILE DIFFERENT ROWS (I) CORRESPOND TO DIFFERENT PHASE   NXMODE
C          VELOCITIES.  THE SEARCH PROCEEDS FROM AN INITIAL POINT (IST,JS NXMODE
C          TO SUCCESSIVE ADJACENT POINTS HAVING THE SAME INMODE AS THE   NXMODE
C          STARTING POINT.  THE DETERMINATION OF (IFND,JFND) IS SUBJECT T NXMODE
C          THE FOLLOWING RULES.                                          NXMODE
C                                                                       NXMODE
C                 1.  IT MUST LIE BELOW OR TO THE LEFT OF A POINT WITH  NXMODE
C                     OPPOSITE INMODE                                   NXMODE
C                                                                       NXMODE
C                 2.  IF MUST BE THE HIGHEST POINT (LOWEST I) IN THE REGIO NXMODE
C                     SATISFYING CONDITION 1                            NXMODE
C                                                                       NXMODE
C                 3.  IF MORE THAN 1 POINT SATISFY 1 AND 2, THEN THE POINT NXMODE
C                     DETERMINED IS THAT FURTHEST TO THE LEFT.          NXMODE
C                                                                       NXMODE
C                 4.  ONLY POINTS IN THE RECTANGLE ARE CONSIDERED       NXMODE
C                                                                       NXMODE
C          THE COMPUTATION ASSUMES REGION OF SUCCESSIVELY ADJACENT POINTS NXMODE
C          HAVING SAME INMODE IS SIMPLY CONNECTED AND THAT PHASE VELOCITY NXMODE
C          CURVES BEND DOWNWARDS, I.E., D(VP)/D(OM) .LT. 0.  (THIS CAN BE NXMODE
C          THE CASE PROVIDING VP IS GREATER THAN THE MAXIMUM WIND       NXMODE
C          VELOCITY.)  IF THE POINT IS FOUND, K=1, IF NOT FOUND, K=-1.  NXMODE
C                                                                       NXMODE
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)           NXMODE
C AUTHOR      - A.D.PIERCE, M.I.T., JUNE,1968.                          NXMODE
C                                                                       NXMODE
C                    ----CALLING SEQUENCE----                           NXMODE
C                                                                       NXMODE
C SEE SUBROUTINE ALLMOD                                                 NXMODE
C     DIMENSION INMODE(1)          (VARIABLE DIMENSIONING)              NXMODE
C     CALL NXMODE(IST,JST,NOM,NVP,INMODE,IFND,JFND,K)                   NXMODE
C                                                                       NXMODE
C NO EXTERNAL SUBROUTINES ARE REQUIRED                                  NXMODE
C                                                                       NXMODE
C                 ----ARGUMENT LIST----                                 NXMODE
C                                                                       NXMODE
C     IST          I*4     NO     INP                                   NXMODE
C     JST          I*4     NO     INP                                   NXMODE
C     NOM          I*4     NO     INP                                   NXMODE
C     NVP          I*4     NO     INP                                   NXMODE
C     INMODE       I*4     VAR    INP                                   NXMODE
C     IFND         I*4     NO     OUT                                   NXMODE
C     JFND         I*4     NO     OUT                                   NXMODE
C     K            I*4     NO     OUT                                   NXMODE
C                                                                       NXMODE
C NO COMMON STORAGE USED                                                NXMODE
C                                                                       NXMODE
C                    ----INPUTS----                                     NXMODE
C                                                                       NXMODE
C     IST         =ROW INDEX OF START POINT                             NXMODE
```

```
C     JST          =COLUMN INDEX OF START POINT
C     NOM          =NO. OF COLUMNS OF ARRAY
C     NVP          =NO. OF ROWS OF ARRAY
C     INMODE(L)    =SIGN OF NORMAL MODE DISPERSION FUNCTION. 1 IF POS.,
C                   -1 IF NEG., 5 IF IT DOESN'T EXIST.  LET I=L MOD NVP,
C                   J=(L-I)/NVP+1.  INMODE(L) IS SIGN OF NMDF FOR
C                   OMEGA=OM(J), PHASE VEL. =VP(I), WHERE OM(J) .GE. OM(
C                   AND VP(I) .LE. VP(I-1).
C
C                   ----OUTPUTS----
C
C     IFND         =ROW INDEX OF FOUND POINT
C     JFND         =COLUMN INDEX OF FOUND POINT
C     K            =FLAG INDICATING IF POINT (IFND,JFND) IS FOUND. 1 IF
C                   YES, -1 IF NO.
C
C                   ----EXAMPLE----
C
C SUPPOSE THE ARRAY OF INMODE VALUES IS AS SHOWN BELOW
C
C                +++++++++--       NVP=8, NOM=11
C                ++++++++++-
C                5------++++     IF IST=8,JST=5  THEN IFND=3,JFND=2,K=1
C                55-----++++     IF IST=2,JST=5  THEN IFND=1,JFND=9,K=1
C                55--------+     IF IST=3,JST=7  THEN IFND=3,JFND=2,K=-1
C                55--------+     IF IST=8,JST=2  THEN K=-1
C                55--------+     IF IST=2,JST=11 THEN K=-1
C                55--------+
C
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C
    1 IF( IST .GT. NVP .OR. JST .GT. NOM) GO TO 100
      J9=(JST-1)*NVP+IST
      I0=INMODE(J9)
    3 IF( I0 .NE. 1 .AND. I0 .NE. -1) GO TO 100
C
C THE POINT (IST,JST) LIES IN THE ARRAY AND THE NORMAL MODE DISPERSION
C FUNCTION EXISTS AT THIS POINT WITH A SIGN I0.  WE FIRST GO UP UNTIL
C A DIFFERENT INMODE IS ENCOUNTERED OR UNTIL WE REACH I=1
      I=IST
      J=JST
   10 IF( I .EQ. 1) GO TO 30
      I=I-1
      J10=(J-1)*NVP+I
      ICHK=INMODE(J10)
      IF( ICHK .EQ. I0) GO TO 10
      I=I+1
C
C THE CURRENT I IS NOT 1.  IF THE ICHK OF THE POINT ABOVE IS NOT 5, WE
C MOVE TO THE LEFT.
   15 IF( ICHK .EQ. 5) GO TO 50
      IF( J .EQ. 1) GO TO 20
      J=J-1
      J10=(J-1)*NVP+I
      ICHK=INMODE(J10)
C
C IF THE ICHK OF THE CONSIDERED NEW POINT IS I0, WE TRY TO GO HIGHER
C AGAIN.
      IF(ICHK .EQ. I0) GO TO 10
      J=J+1
C
```

```
C WE HAVE -IO ABOVE THE CURRENT POINT AND ARE EITHER ON THE FAR LEFT OF NXMODE
C THE TABLE OR ELSE HAVE A DIFFERENT SIGN AT THE POINT TO THE LEFT.     NXMODE
C THIS IS INTERPRETED AS SUCCESS.                                       NXMODE
   20 K=1                                                               NXMODE
      IFND=I                                                            NXMODE
      JFND=J                                                            NXMODE
      RETURN                                                            NXMODE
C                                                                       NXMODE
C THE CONSIDERED NEW POINT IS ON THE FIRST ROW.  WE GO TO THE RIGHT.    NXMODE
   30 IF( J .EQ. NOM) GO TO 60                                          NXMODE
      J=J+1                                                             NXMODE
      J10=(J-1)*NVP+I                                                   NXMODE
      ICHK=INMODE(J10)                                                  NXMODE
      IF( ICHK .EQ. IO) GO TO 30                                        NXMODE
      J=J-1                                                             NXMODE
C                                                                       NXMODE
C IF THE POINT AT THE RIGHT OF CURRENT (I,J) IS -IO, WE HAVE SUCCESS    NXMODE
      IF( ICHK .EQ. -IO) GO TO 20                                       NXMODE
C                                                                       NXMODE
C IF IT IS NOT -IO, WE ALLOW FOR POSSIBILITY OF INMODES=5 IN UPPER RIGH NXMODE
C HAND CORNER OF THE TABLE AND TRY TO SKIRT THESE FIVES BY MOVING EITHE NXMODE
C DOWNWARDS OR TO THE RIGHT.                                            NXMODE
   40 IF( I .EQ. NVP) GO TO 70                                          NXMODE
      I=I+1                                                             NXMODE
      J10=(J-1)*NVP+I                                                   NXMODE
      ICHK=INMODE(J10)                                                  NXMODE
C                                                                       NXMODE
C IF THIS ICHK IS +IO WE ARE IN A POSITION TO MAKE A TRY OF MOVING TO   NXMODE
C THE RIGHT.                                                            NXMODE
   44 IF( ICHK .NE. IO) GO TO 90                                        NXMODE
C                                                                       NXMODE
C IF WE ARE ON THE RIGHT HAND SIDE OF THE TABLE THE DESIRED POINT CANNO NXMODE
C BE FOUND. WE RETURN WITH K=-1                                         NXMODE
   45 IF( J .EQ. NOM) GO TO 100                                         NXMODE
      J=J+1                                                             NXMODE
C                                                                       NXMODE
C IT IS TAKEN FOR GRANTED THAT THE INMODE OF POINT ABOVE CURRENT (I,J)  NXMODE
C IS 5 SINCE IT HAS FOUND TO BE 5 TO THE LEFT AND ABOVE.  THE INMODE OF NXMODE
C THE POINT TO THE LEFT IS 10.  IF THE NEW INMODE IS +IO, WE HAVE TO TR NXMODE
C TO MOVE FURTHER TO THE RIGHT.                                         NXMODE
      J10=(J-1)*NVP+I                                                   NXMODE
      ICHK=INMODE(J10)                                                  NXMODE
      IF( ICHK .EQ. IO ) GO TO 45                                       NXMODE
      J=J-1                                                             NXMODE
C                                                                       NXMODE
C IF THE CURRENT ICHK IS 5, WE TRY TO GO DOWN AGAIN.  THE OTHER POSS-   NXMODE
C IBILITY, ICHK=-IO INDICATES SUCCESS                                   NXMODE
      IF( ICHK .EQ. -IO) GO TO 20                                       NXMODE
      GO TO 40                                                          NXMODE
C                                                                       NXMODE
C WE CONTINUE HERE FROM 15.  THE POINT ABOVE THE CURRENT (I,J) HAS      NXMODE
C ICHK .EQ. 5.  THE SITUATION IS SUCH THAT WE CAN RESUME CALCULATION    NXMODE
C AT 45 AND TRY TO MOVE FURTHER TO THE RIGHT.                           NXMODE
   50 GO TO 45                                                          NXMODE
C                                                                       NXMODE
C WE CONTINUE HERE WITH I=1,J=NOM FROM STATEMENT 30.  SINCE WE HAVE NO  NXMODE
C PLACE TO GO THE SEARCH IS UNSUCCESSFUL.  WE RETURN WITH K=-1.         NXMODE
   60 GO TO 100                                                         NXMODE
C                                                                       NXMODE
C WE CONTINUE HERE FROM STATEMENT 40 WITH I .EQ. NVP AND INMODE=5 TO TH NXMODE
C RIGHT OF THE CURRENT (I,J).  WE RETURN WITH K=-1.                     NXMODE
   70 GO TO 100                                                         NXMODE
C                                                                       NXMODE
C WE CONTINUE HERE FROM STATEMENT 44 WITH THE POINT BELOW HAVING        NXMODE
```

```
C ICHK .NE. I0.  THE POINT AT THE RIGHT HAS ICHK .EQ. 5.  WE CANNOT     NXMODE
C SKIRT THE FIVES AND HENCE WE RETURN WITH K=-1.                        NXMODE
   80 GO TO 100                                                         NXMODE
C                                                                       NXMODE
C WE CONTINUE HERE FROM 1,3,45,60,70,OR 80.  THE SEARCH WAS UNSUCCESSFU NXMODE
  100 K=-1                                                              NXMODE
      RETURN                                                            NXMODE
      END                                                               NXMODE
```

```
      SUBROUTINE NXTPNT(I1,J1,ITYP1,I2,J2,ITYP2,NROW,NCOL,INM,K)         NXTPNT
C     NXTPNT (SUBROUTINE)                  6/24/69   LAST CARD IN DECK IS NXTPNT
C                                                                         NXTPNT
C                                                                         NXTPNT
C                 ----ABSTRACT----                                        NXTPNT
C                                                                         NXTPNT
C TITLE - NXTPNT                                                          NXTPNT
C         PROGRAM TO FIND THE NEXT POINT (I2,J2) OF AN ARRAY OF NROW ROW  NXTPNT
C         AND NCOL COLUMNS GIVEN THE PRECEDING POINT (I1,J1).  POINT WIL  NXTPNT
C         BE USED IN SUBSEQUENT CALCULATION OF A PARTICULAR POINT ON THE  NXTPNT
C         PHASE VELOCITY VERSUS FREQUENCY CURVE OF A GIVEN GUIDED MODE.   NXTPNT
C         A TABLE OF VALUES OF THE SIGN OF THE NORMAL MODE DISPERSION     NXTPNT
C         FUNCTION IS PRESUMED TO BE STORED AS INM((J-1)*NVP+I) FOR EACH  NXTPNT
C         POINT (I,J) IN THE ARRAY.  DIFFERENT COLUMNS (J) CORRESPOND TO  NXTPNT
C         DIFFERENT FREQUENCIES WHILE DIFFERENT ROWS (I) CORRESPOND TO    NXTPNT
C         DIFFERENT PHASE VELOCITIES.  SUCCESSIVE POINTS ARE CHARACTERIZ  NXTPNT
C         BY A TYPE, ITYP1 IS TYPE OF (I1,J1) WHILE ITYP2 IS TYPE OF      NXTPNT
C         SECOND POINT.  THE TYPE INDEX IS 1 IF THE POINT DIRECTLY ABOVE  NXTPNT
C         THE CONSIDERED POINT HAS AN INM OF OPPOSITE SIGN.  IT IS 2 IF   NXTPNT
C         THE POINT TO THE RIGHT HAS INM OF OPPOSITE SIGN.  SINCE BOTH    NXTPNT
C         POSSIBILITIES CAN OCCUR, THE DESIGNATED TYPE INDEX ITYP1 DENOT  NXTPNT
C         THE PREVIOUS USE OF THE POINT (I1,J1) IN COMPUTATION.  THE VAL  NXTPNT
C         ITYP2 WILL IN GENERAL DEPEND ON THE PREVIOUS VALUE ITYP1.       NXTPNT
C         THE DERIVED VALUES OF I2,J2,ITYP2 ARE CALCULATED AS FOLLOWS.    NXTPNT
C                                                                         NXTPNT
C                1.  IF ITYP1 IS 1 AND INM OF POINT TO RIGHT IS OPPOSITE  NXTPNT
C                    OF I0=INM((J-1)*NVP+I), THEN I2=I1,J2=J1,ITYP2=2.    NXTPNT
C                                                                         NXTPNT
C                2.  THE POINT (I2,J2) MUST EITHER BE THE DIRECTLY ADJACE NXTPNT
C                    POINT TO THE RIGHT (I1,J1+1), THE POINT DIRECTLY BEL NXTPNT
C                    (I1+1,J1), OR THE ADJACENT POINT TO THE LOWER RIGHT  NXTPNT
C                    (I1+1,J1+1) IF CONDITION 1 DOES NOT HOLD             NXTPNT
C                                                                         NXTPNT
C                3.  THE CHOSEN POINT MUST HAVE THE SAME INM AS (I1,J1)   NXTPNT
C                    AND HAVE A POINT EITHER DIRECTLY ABOVE OR DIRECTLY T NXTPNT
C                    THE RIGHT WITH OPPOSITE INM.                         NXTPNT
C                                                                         NXTPNT
C                4.  IN THE EVENT MORE THAN ONE POINT SATISFY CONDITIONS  NXTPNT
C                    2 AND 3, PRIORITY OF SELECTION IS (1) THE POINT TO   NXTPNT
C                    THE RIGHT, (2) THE POINT DIRECTLY BELOW, (3) THE POI NXTPNT
C                    TO THE LOWER RIGHT.  IF THE SELECTED POINT SATISFIES NXTPNT
C                    CRITERIA FOR BOTH ITYP2=1 OR 2, ITYP2=1 IS RETURNED. NXTPNT
C                    OTHERWISE, THE APPROPRIATE ITYP2 IS RETURNED DEPENDI NXTPNT
C                    ON WHICH CRITERION IS SATISFIED.                     NXTPNT
C                                                                         NXTPNT
C         THE COMPUTATION ASSUMES REGION OF SUCCESSIVELY ADJACENT POINTS  NXTPNT
C         HAVING SAME INM TO BE SIMPLY CONNECTED AND THAT PHASE VELOCITY  NXTPNT
C         CURVES BEND DOWNWARDS, I.E.,  D(VP)/D(OM) .LT. 0.  IF NEW POIN  NXTPNT
C         IS FOUND, K=+1.  IF IT IS NOT FOUND, K=-1.                      NXTPNT
C                                                                         NXTPNT
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             NXTPNT
C AUTHOR      - A.D.PIERCE, M.I.T., JUNE,1968                             NXTPNT
C                                                                         NXTPNT
C                 ----CALLING SEQUENCE----                                NXTPNT
C                                                                         NXTPNT
C SEE SUBROUTINE MODETR                                                   NXTPNT
C     DIMENSION INMODE(1)          (INMODE IS SAME AS INM)                NXTPNT
C     CALL NXTPNT(I1,J1,ITYP1,I2,J2,ITYP2,NROW,NCOL,INMODE,K)             NXTPNT
C     IF( K .EQ. -1) GO SOMEWHERE                                         NXTPNT
C     USE I2,J2,ITYP2                                                     NXTPNT
C                                                                         NXTPNT
C NO EXTERNAL LIBRARY SUBROUTINES ARE REQUIRED                            NXTPNT
C                                                                         NXTPNT
C              ----ARGUMENT LIST----                                      NXTPNT
```

```
C
C      I1          I*4     NO     INP
C      J1          I*4     NO     INP
C      ITYP1       I*4     NO     INP
C      I2          I*4     NO     OUT
C      J2          I*4     NO     OUT
C      ITYP2       I*4     NO     OUT
C      NROW        I*4     NO     INP
C      NCOL        I*4     NO     INP
C      INM         I*4     VAR    INP
C      K           I*4     NO     OUT
C
C NO COMMON STORAGE USED
C
C                    ----INPUTS----
C
C      I1           =ROW INDEX OF START POINT
C      J2           =COLUMN INDEX OF START POINT
C      ITYP1        =TYPE INDEX OF START POINT, 1 MEANS POINT ABOVE HAS
C                    DIFFERENT INM, 2 MEANS POINT TO RIGHT HAS DIFFERENT
C                    INM.
C      NROW         =NUMBER OF ROWS IN ARRAY
C      NCOL         =NUMBER OF COLUMNS IN ARRAY
C      INM          =SIGN OF NORMAL MODE DISPERSION FUNCTION, 1 IF POS.,
C                    -1 IF NEG., 5 IF IT DOESN'T EXIST.  LET I=L MOD NVP,
C                    J=(L-I)/NVP+1.  INMCODE(L) IS SIGN OF NMDF FOR
C                    OMEGA=OM(J), PHASE VEL. =VP(I), WHERE OM(J) .GE. OM(
C                    AND VP(I) .LE. VP(I-1)
C
C                    ----OUTPUTS----
C
C      I2           =ROW INDEX OF FOUND POINT
C      J2           =COLUMN INDEX OF FOUND POINT
C      ITYP2        =TYPE INDEX OF FOUND POINT
C      K            =FLAG INDICATING IF POINT (I2,J2) IS FOUND, 1 IF YES,
C                    -1 IF NO
C
C                    ----EXAMPLE----
C
C SUPPOSE THE ARRAY OF INM VALUES IS AS SHOWN BELOW
C
C                  +++++++++--       NROW=8, NCOL=11
C                  +++++++++--
C                  5-------+++    IF I1=3,J1=4,ITYP1=1 THEN I2=3,J2=5,
C                  55------+++       ITYP2=1,K=1
C                  55--------+
C                  55--------+    IF I1=1,J1=9,ITYP1=2  THEN I2=2,J2=10,
C                  55--------+       ITYP2=1,K=1
C                  55--------+
C                                 IF I1=3,J1=7,ITYP1=1  THEN I2=3,J2=7,
C                                    ITYP2=2,K=1
C
C                                 IF I1=3,J1=11,ITYP1=1 THEN K=-1
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION INM(1)
      J11=(J1-1)*NROW+I1
      I0=INM(J11)
    1 IF( I0 .EQ. 5 .OR. I1 .GT. NROW .OR. J1 .GE. NCOL) GO TO 30
C
```

```
C IR IS INM OF POINT TC THE RIGHT.  IO IS INM OF POINT (I1,J1).        NXTPNT
    5 J12=(J1)*NROW+I1                                                  NXTPNT
      IR=INM(J12)                                                       NXTPNT
    6 IF( IR .NE. IO ) GO TO 15                                         NXTPNT
    7 IF( I1 .EQ. 1 ) GO TO 30                                          NXTPNT
C                                                                       NXTPNT
C IR HAS THE SAME SIGN AS IO.  WE CHECK IRU REPRESENTING INM OF UPPER   NXTPNT
C RIGHT POINT.  IF THIS IS -IO, THE RIGHT POINT IS THE DESIRED POINT.   NXTPNT
C IF IT IS NOT -IO, WE CANNOT FIND (I2,J2).                             NXTPNT
   10 J13=(J1)*NROW+I1-1                                                NXTPNT
      IRU=INM(J13)                                                      NXTPNT
   11 IF( IRU .NE. -IO ) GO TO 30                                       NXTPNT
      ITYP2=1                                                           NXTPNT
      I2=I1                                                             NXTPNT
      J2=J1+1                                                           NXTPNT
      K=1                                                               NXTPNT
      RETURN                                                            NXTPNT
C                                                                       NXTPNT
C WE ARRIVE HERE FROM STATEMENT 6.  THE POINT TO THE RIGHT HAS A        NXTPNT
C DIFFERENT INM.  IF THIS IS -IO AND ITYP1=1, WE HAVE (I2,J2)=(I1,J1)   NXTPNT
C WITH ITYP2=2.  IF THIS IS 5, WE CANNOT FIND (I2,J2).                  NXTPNT
   15 IF( IR .EQ. 5 ) GO TO 30                                          NXTPNT
C                                                                       NXTPNT
C IR=-IO AT THIS POINT                                                  NXTPNT
      IF( ITYP1 .NE. 1 ) GO TO 25                                       NXTPNT
      I2=I1                                                             NXTPNT
      J2=J1                                                             NXTPNT
      ITYP2=2                                                           NXTPNT
      K=1                                                               NXTPNT
      RETURN                                                            NXTPNT
C                                                                       NXTPNT
C IR=-IO.  ITYP1 IS 2.  WE CONTINUE FROM STATEMENT 15.  IF WE ARE ON TH NXTPNT
C BOTTOM ROW, WE CANNOT FIND NEW POINT                                  NXTPNT
   25 IF (I1.EQ.NROW) GO TO 30                                          NXTPNT
C                                                                       NXTPNT
C WE CONSIDER POINTS BELOW AND TO LOWER RIGHT                           NXTPNT
      J14=(J1-1)*NROW+I1+1                                              NXTPNT
      IO=INM(J14)                                                       NXTPNT
      J15=(J1)*NROW+I1+1                                                NXTPNT
      IOR=INM(J15)                                                      NXTPNT
C                                                                       NXTPNT
C IF IOR IS 5, WE CANNOT FIND THE NEW POINT                             NXTPNT
   26 IF( IOR .EQ. -5 ) GO TO 30                                        NXTPNT
C                                                                       NXTPNT
C IF IOR IS IO, THE NEXT POINT IS THE DR POINT                          NXTPNT
   27 IF( IOR .NE. IO ) GO TO 28                                        NXTPNT
      I2=I1+1                                                           NXTPNT
      J2=J1+1                                                           NXTPNT
      ITYP2=1                                                           NXTPNT
      K=1                                                               NXTPNT
      RETURN                                                            NXTPNT
C                                                                       NXTPNT
C IR=-IO, ITYP1 IS 2, IOR IS -IO.  WE CONTINUE FROM STATEMENT 27.       NXTPNT
   28 IF( IO .NE. IO ) GO TO 30                                         NXTPNT
C                                                                       NXTPNT
C THE NEXT POINT IS THE DOWN POINT                                      NXTPNT
      I2=I1+1                                                           NXTPNT
      J2=J1                                                             NXTPNT
      ITYP2=2                                                           NXTPNT
      K=1                                                               NXTPNT
      RETURN                                                            NXTPNT
C                                                                       NXTPNT
C WE ARRIVE HERE FROM 1,7,11,15,25,26.  THE NEXT POINT CANNOT BE FOUND  NXTPNT
   30 K=-1                                                              NXTPNT
      RETURN                                                            NXTPNT
      END                                                               NXTPNT
```

```
      SUBROUTINE PAMPDE(ZSCRCE,ZOBS,MOFNO,KST,KFIN,OMMOD,VPMOD,AKI,       PAMPDE
     1AMP,ALAM,FACT,THETK,NPRNT)                                           PAMPDE
C     PAMPDE (SUBROUTINE)               7/30/66   LAST CARD IN DECK IS  PAMPDE
C                                                                       PAMPDE
C                  ----ABSTRACT----                                     PAMPDE
C                                                                       PAMPDE
C TITLE - PAMPDE                                                        PAMPDE
C        PROGRAM TO COMPUTE AND STORE AMPLITUDE FACTORS AMP(J) AND FACT PAMPDE
C        AND SCALING FACTOR ALAM.  THE QUANTITY AMP(J) IS THE QUANTITY  PAMPDE
C        AMPLTD COMPUTED BY SUBROUTINE NAMPDE WHEN THE ANGULAR FRECUENC PAMPDE
C        IS OMMOD(J) AND THE PHASE VELOCITY IS VPMOD(J).  IT CORRESPOND PAMPDE
C        TO THE NMODE-TH GUIDED MODE WHEN J IS BETWEEN KST(NMODE) AND   PAMPDE
C        KFIN(NMODE), INCLUSIVE.  THE QUANTITY FACT IS DEPENDENT ON     PAMPDE
C        SOURCE ALTITUDE ZSCRCE AND OBSERVER ALTITUDE ZOBS AND IS GIVEN PAMPDE
C                                                                       PAMPDE
C          FACT = CONST*CI(1)*UEC*(PSCRCE/1.E6)**0.333333               PAMPDE
C                                                                       PAMPDE
C        WHERE CONST=4.3/SQRT(2*PI), CI(1) IS THE SOUND SPEED AT THE    PAMPDE
C        GROUND, (PSCRCE/1.E6) IS THE AMBIENT PRESSURE AT ZSCRCE DIVIDE PAMPDE
C        BY THE AMBIENT PRESSURE AT THE GROUND.  THE QUANTITY UED IS    PAMPDE
C        THE SQUARE ROOT OF (AMBIENT DENSITY AT ZOBS)/(AMBIENT DENSITY  PAMPDE
C        ZSCRCE ).  THE SCALING FACTOR ALAM IS GIVEN BY                 PAMPDE
C                                                                       PAMPDE
C          ALAM = (1.E6/PSCRCE)**(0.333333)*(CI(1)/CI(ISCR)             PAMPDE
C                                                                       PAMPDE
C        WHERE CI(ISCR) IS THE SOUND SPEED AT THE SOURCE ALTITUDE.  THE PAMPDE
C        SIGNIFICANCE OF THESE QUANTITIES IS EXPLAINED IN SUBROUTINE    PAMPDE
C        PPAMP.                                                         PAMPDE
C                                                                       PAMPDE
C PROGRAM NOTES                                                         PAMPDE
C                                                                       PAMPDE
C              THE PARAMETERS IMAX,CI,VXI,VYI,HI DEFINING THE MULTILAYE PAMPDE
C              ATMOSPHERE ARE PRESUMED STORED IN COMMON.  THE AMBIENT   PAMPDE
C              PRESSURES ARE COMPUTED BY CALLING SUBROUTINE AMBNT WHICH PAMPDE
C              ALSO COMPUTES THE INDICES IOBS AND ISCR OF THE LAYERS    PAMPDE
C              IN WHICH OBSERVER AND SOURCE, RESPECTIVELY, LIE.         PAMPDE
C                                                                       PAMPDE
C              IN COMPUTING AMBIENT DENSITIES, THE IDEAL GAS LAW         PAMPDE
C              RHO= GAMMA*P/C**2 IS USED. THUS UED = (CI(ISCR)/CI(IOBS) PAMPDE
C              SQRT(POBS/PSCRCE).                                       PAMPDE
C                                                                       PAMPDE
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)           PAMPDE
C                                                                       PAMPDE
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JULY,1968               PAMPDE
C                                                                       PAMPDE
C                  ----CALLING SEQUENCE----                             PAMPDE
C                                                                       PAMPDE
C SEE THE MAIN PROGRAM                                                  PAMPDE
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                       PAMPDE
C     DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),AMP(1)                 PAMPDE
C THE PROGRAM USES VARIABLE DIMENSIONING FOR QUANTITIES IN ITS          PAMPDE
C ARGUMENT LIST.                                                        PAMPDE
C     COMMON IMAX,CI,VXI,VYI,HI  THESE MUST BE STORED IN COMMON)        PAMPDE
C     CALL PAMPDE(ZSCRCE,ZOBS,MOFNO,KST,KFIN,OMMOD,VPMOD,AMP,ALAM,      PAMPDE
C    1 FACT,THETK,NPRNT)                                                PAMPDE
C                                                                       PAMPDE
C                  ----EXTERNAL SUBROUTINES REQUIRED----                PAMPDE
C                                                                       PAMPDE
C     AMBNT,NAMPDE,TCTINT,MMPM,AAAA,USEAS,UPINT,ELINT,B088,CAI,SAI      PAMPDE
C                                                                       PAMPDE
C                  ----ARGUMENT LIST----                                PAMPDE
C                                                                       PAMPDE
C     ZSCRCE      R*4    ND     INP                                     PAMPDE
C     ZOBS        R*4    ND     INP                                     PAMPDE
```

```
C     MOFNO       I*4    NO    INP
C     KST         I*4    VAR   INP
C     KFIN        I*4    VAR   INP
C     OMMOD       R*4    VAR   INP
C     VPMOD       R*4    VAR   INP
C     AMP         R*4    VAR   OUT
C     ALAM        R*4    NO    OUT
C     FACT        R*4    NO    OUT
C     THETK       R*4    NO    INP
C     NPRNT       I*4    NO    INP
C
C COMMON STORAGE USED
C     COMMON IMAX,CI,VXI,VYI,HI
C
C     IMAX        I*4    NO    INP
C     CI          R*4    100   INP
C     VXI         R*4    100   INP
C     VYI         R*4    100   INP
C     HI          R*4    100   INP
C
C                  ----INPUTS----
C
C     ZSCRCE      =HEIGHT IN KM OF BURST ABOVE GROUND
C     ZOBS        =HEIGHT IN KM OF OBSERVER ABOVE GROUND
C     MOFNO       =NUMBER OF GUIDED MODES FOUND
C     KST(N)      =INDEX OF FIRST TABULATED POINT IN N-TH MODE
C     KFIN(N)     =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN
C                  GENERAL, KFIN(N)=KST(N+1)-1.
C     OMMOD(N)    =ARRAY STORING ANGULAR FREQUENCY ORDINATE (RAD/SEC) O
C                  POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR
C                  FOR N BETWEEN KST(NMODE) AND KFIN(NMODE), INCLUSIVE.
C     VPMOD(N)    =ARRAY STORING PHASE VELOCITY ORDINATE (KM/SEC) OF
C                  POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR
C                  FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).
C     THETK       =DIRECTION IN RADIANS TO OBSERVER FROM SOURCE, RECKON
C                  COUNTER CLOCKWISE FROM X AXIS.
C     NPRNT       =PRINT OPTION INDICATOR (SEE NAM1 IN MAIN PROGRAM).
C     IMAX        =NUMBER OF LAYERS OF FINITE THICKNESS.
C     CI(I)       =SOUND SPEED IN KM/SEC IN I-TH LAYER
C     VXI(I)      =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)
C     VYI(I)      =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)
C     HI(I)       =THICKNESS IN KM OF I-TH LAYER OF FINITE THICKNESS
C
C                  ----OUTPUTS----
C
C     AMP(J)      =AMPLITUDE FACTOR FOR GUIDED WAVE EXCITED BY POINT
C                  ENERGY SOURCE.  UNITS ARE KM**(-1).  THE J-TH ELEMEN
C                  CORRESPONDS TO ANGULAR FREQUENCY OMMOD(J) AND PHASE
C                  VELOCITY VPMOD(J).  THE AMPLITUDE FACTOR IS APPROPRI
C                  TO THE NMODE-TH MODE IF J .GE. KST(NMODE) AND J .LE.
C                  KFIN(NMODE).  THE AMP(J) IS THE SAME AS AMPLTD COMPU
C                 BY SUBROUTINE NAMPOE.
C     ALAM        =A SCALING FACTOR DEPENDENT ON HEIGHT OF BURST, EQUAL
C                  TO CUBE ROOT OF (PRESSURE AT GROUND)/(PRESSURE AT
C                  BURST HEIGHT) TIMES (SOUND SPEED AT GROUND)/(SOUND
C                  SPEED AT BURST HEIGHT).
C     FACT        =A GENERAL AMPLITUDE FACTOR DEPENDENT ON BURST HEIGHT
C                  AND OBSERVER HEIGHT.  A PRECISE DEFINITION IS GIVEN
C                  IN THE ABSTRACT.
C
C                  ----PROGRAM FOLLOWS BELOW----
C
C
C DIMENSION AND COMMON STATEMENTS
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
```

```
      DIMENSION KST(10),KFIN(10),OMMOD(1000),VPMOD(1000),AMP(1000)
      DIMENSION AKI(1000)
      COMMON IMAX,CI,VXI,VYI,HI
C
      MOFNO = MOFNO
      IF (NPRNT.LT.0) GO TO 20
C PRINT HEADING FOR PHI1 AND PHI2 PROFILE DATA TO BE PRINTED BY NAMPDE
      WRITE (6,19)
   19 FORMAT (1H1,41X,26HPHI1 AND PHI2 PROFILE DATA ///63HOIAP1MX = NO.
     1OF LAYER FOR WHICH ABS(PHI1(IAP1MX)) IS A MAXIMUM/63H IAP2MX = NO
     2 OF LAYER FOR WHICH ABS(PHI2(IAP2MX)) IS A MAXIMUM/42H R1      = P
     3I1(IAP1MX) / ABS(PHI2(IAP2MX)) /42H R2      = PHI2(IAP2MX) / ABS(P
     4I2(IAP2MX)) /37H R3      = PHI2(1) / ABS(PHI2(IAP2MX)) /40H NZC1
     5= NO. OF TIMES PHI1 CHANGES SIGN /40H NZC2    = NO. OF TIMES PHI2
     6HANGES SIGN)
   20 CONTINUE
C DO LOOP TO COMPUTE AMP(J)
      DO 25  II=1,MOFNO
      IF (NPRNT.LT.0) GO TO 23
      WRITE (6,22) II
   22 FORMAT (1H ////// 1H ,51X,5HMODE ,I2 /// 1H ,7X,5HOMEGA,7X,5HVPHS
     1,6X,6HIAP1MX,10X,2HR1,8X,4HNZC1,6X,6HIAP2MX,10X,2HR2,8X,4HNZC2,10
     2,2HR3 /)
   23 J1=KST(II)
      J2=KFIN(II)
      DO 25  J=J1,J2
      K = J
      OMEGA = OMMOD(K)
      VPHSE = VPMOD(K)
      AKITR = AKI(K)
      X = AMP(K)
      CALL NAMPDE(ZSCRCE,ZOBS,OMEGA,VPHSE,AKITR,THETK,X,NPRNT)
      AMP(K) = X
   25 CONTINUE
      WRITE (6,251)
  251 FORMAT(1H ///// 16H AMBNT IS CALLED)
C END OF DO LOOP
C
C COMPUTATION OF AMBIENT PRESSURES
      CALL AMBNT(ZSCRCE,PSCRCE,ISCR)
      CALL AMBNT(ZOBS,POBS,ICBS)
      WRITE (6,252) PSCRCE,ISCR,POBS,ICBS
  252 FORMAT (1H ,E16.5,I10,E16.5,I10)
C
C COMPUTATION OF SQRT(DENSITY RATIO)
      UEO = (CI(ISCR)/CI(ICBS)) * SQRT(POBS/PSCRCE)
C
C COMPUTATION OF ALAM AND FACT
      ALAM=(1.E6/PSCRCE)**(0.333333)*(CI(1)/CI(ISCR))
C NOTE THAT CI(1) IS SOUND SPEED AT THE GROUND
      CONST = 4.0/SQRT(2.0*3.141593)
      FACT = CONST*(CI(1)*UEO*(PSCRCE/1.E6)**(0.3333333))
      WRITE (6,253) FACT
  253 FORMAT (1H ,5HFACT=,E16.5)
      IF(NPRNT .NE. 1) RETURN
      WRITE (6,31) ZSCRCE,ZOBS,FACT,ALAM
   31 FORMAT(1H1, 20X, 36HTABULATION OF SOURCE FREE AMPLITUDES,
     1 23H FROM SUBROUTINE PAMPDE //21X, 19HHEIGHT OF BURST    =,
     1 F8.3, 3H KM / 21X, 19HHEIGHT OF OBSERVER=, F8.3, 3H KM/
     1 21X, 4HFACT, 14X, 1H=, F8.3, 7H KM/SEC/ 21X,4HALAM,14X, 1H=,
     1 F8.3)
      DO 50 II =1,MOFNO
      WRITE (6,41) II
   41 FORMAT( 1H /// 1H , 5HMODE , I3/ 1H , 20X,5HOMEGA,
     1 15X, 5HVPHSE,15X, 3HAKI, 17X, 3HAMP)
```

```
      K1=KST(II)
      K2=KFIN(II)
      DO 50 J=K1,K2
   50 WRITE (6,51) OMMOD(J),VPMOD(J),AKI(J),AMP(J)
   51 FORMAT(1H ,4X,F20.5,F20.5,F20.9,F20.5)
      RETURN
      END
```

```
      SUBROUTINE PHASE(RR,RI,R,PHI)
C     PHASE (SUBROUTINE)                   8/15/68   LAST CARD IN DECK IS
C
C
C                   ----ABSTRACT----
C
C TITLE - PHASE
C          CONVERSION OF A COMPLEX NUMBER FROM RECTANGULAR FORM TO POLAR
C          FORM
C
C                GIVEN TWO REAL NUMBERS RR AND RI, A MAGNITUDE R AND AN
C                ANGLE PHI ARE COMPUTED SUCH THAT
C
C                     RR + I*RI = R * EXP( I*PHI )
C
C                WHERE I = (-1)**0.5  .
C
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., AUGUST,1968
C
C
C                   ----USAGE----
C
C     NO SUBROUTINES ARE CALLED
C
C FORTRAN USAGE
C
C     CALL PHASE(RR,RI,R,PHI)
C
C INPUTS
C
C    RR        REAL PART OF THE COMPLEX NUMBER BEING CONVERTED
C      R*4
C
C    RI        IMAGINARY PART OF COMPLEX NUMBER BEING CONVERTED
C      R*4
C
C OUTPUTS
C
C    R         MAGNITUDE OF THE COMPLEX NUMBER
C      R*4
C
C    PHI       PHASE OF THE COMPLEX NUMBER (RADIANS) (-PI.LT.PHI.LE.PI)
C      R*4
C
C
C                   ----EXAMPLES----
C
C     CALL PHASE(0.0,1.0,R,PHI)
C
C                R = 1.0  AND  PHI = 1.570796  ARE RETURNED
C
C     CALL PHASE(1.0,-1.0,R,PHI)
C
C                R = 1.414214  AND  PHI = -0.7853982  ARE RETURNED
C
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C
      Q=ABS(RR)+ABS(RI)
      IF(Q-1.E-25) 1,1,30
```

```
    1 R=0.0
      PHI=0.0
      RETURN
   30 AR=RR/Q
      AI=RI/Q
      A=SQRT(AR**2+AI**2)
      R=Q*A
      PHI=ASIN(ABS(AI)/A)
      IF(RR) 50,60,60
   50 IF(RI) 300,300,200
   60 IF(RI) 400,400,100
  100 PHI=PHI
      RETURN
  200 PHI=3.1415927-PHI
      RETURN
  300 PHI=PHI-3.1415927
      RETURN
  400 PHI=-PHI
      RETURN
      END
```

```
      SUBROUTINE PPAMP(YIELC,MCFND,KST,KFIN,OMMOD,VFMOD,
     1AMP,ALAM,FACT,AMPLTO,PHASQ)
C     PPAMP (SUBROUTINE)                 7/30/68   LAST CARD IN DECK IS
C
C                 ----ABSTRACT----
C
C TITLE - PPAMP
C          PROGRAM TO COMPUTE ANC STORE AMPLITUDE ARRAY AMPLTO AND PHASE
C.         ARRAY PHASQ FOR GUIDED WAVES EXCITED BY A POINT ENERGY SOURCE
C          WITH TIME DEPENDENCE CORRESPONDING TO A NUCLEAR EXPLOSION OF
C          ENERGY DENOTEC BY YIELD IN KT.  THE VALUES FOUND ARE TO BE
C          SUBSEQUENTLY USEC BY THPT ACCCROING TO THE RELATION
C
C          (PRESSURE IN DYNES/CM**2 FOR A GIVEN MODE)*SORT(R)
C
C             = INTEGRAL OVER CMEGA OF AMPLTO*COS(OMEGA*(T-R/VP)+PHASQ)
C
C          THE QUANTITIES AMPLTC AND PHASQ ARE BOTH DEPENDENT ON ANGULAR
C          FREQUENCY AND ARE DIFFERENT FOR DIFFERENT MODES.
C
C PROGRAM NOTES
C
C                IN THE FORMULATION FOR A POINT ENERGY SOURCE, THE ENERGY
C                EQUATION IS WRITTEN
C
C                 DP/DT -(C**2)D(RHO)/DT = 4*PI*C**2*F(T)*(DELTA FNCTN )
C                AN EXPRESSION FOR F(T) IS
C
C                 F(T) =((L**2)/CS)*POS*(INTEGRAL OVER X FROM 0 TO CS*T/L
C                        OF UNIVERSAL FUNCTION FUNIV(X))
C
C                WITH L=(ENERGY/POS)**(1/3) AND POS,CS REPRESENTING PRESS
C                AND SOUND SPEED AT THE SOURCE.  IF FIKT(T) IS THE PRESSU
C                AT A DISTANCE OF 1 KM FROM A 1 KT EXPLOSION AT SEA LEVEL
C                AND WITH TIME ORIGIN CORRESPONDING TO BLAST WAVE ONSET,
C                THEN
C
C                 FUNIV(X)=((L1*P01)**(-1))*FIKT(L1*X/C1)
C
C                THE FOURIER TRANSFORM OF F(T) IS ACCORDINGLY FOUND TO BE
C
C                 G(OMEGA)= (1/(2*PI))*(Y**(2/3))*(C1/CS)*(POS/P01)**(1/3
C
C                  *(1/(-I*OMEGA))*FTMAG(OMERAT)*EXP(I*FTPHSE(OMERAT))
C
C                WHERE Y IS YIELD IN KT, I=SQRT(-1), AND OMERAT=ALAM*
C                OMEGA*Y**(1/3).  THE FUNCTIONS FTMAG AND FTPHSE ARE AS
C                COMPUTEC BY SUBROUTINE SOURCE.  THE QUANTIYY ALAM IS
C                (C1/CS)*(P01/POS)**1/3 AS COMPUTED BY SUBROUTINE
C                PAMPOE.
C                A LENGTHY DERIVATION NOT GIVEN HERE INDICATES THAT
C
C                  AMPLTO*EXP(-I*PHASQ)
C
C                    = -4*SQRT(K)*G(OMEGA)*CS*UED*SQRT(2*PI)*AMP
C                      *EXP(-I*PI/4)
C
C                WHERE AMP IS THE SAME AS THE AMPLTD COMPUTED BY NAMPDE A
C                WHERE UED IS THE DENSITY FACTOR (CS/COBS)*SQRT(PSCRCE/PO
C                COMPUTEC IN SUBROUTINE PAMPOE.  INSERTING G(OMEGA) INTO
C                THE ABOVE, WE IDENTIFY
C
C                  PHASQ =(3/4)*PI - FTPHSE(OMERAT)
C
```

```
C                 AMPLTD=FACT*AMP*(Y**(2/3))*FTMAG(OMERAT)*SQRT(K)/OMEGA
C
C               WHERE FACT IS 4/SQRT(2*PI)*C1*UED*(PS/P1)**(1/3) AND IS
C               COMPUTED BY SUBROUTINE PAMPDE.
C
C               THE QUANTITIES FACT, ALAM, AND AMP ARE IN THE INPUT LIST
C               OF THE SUBROUTINE.  NOTE THAT THESE ARE YIELD INDEPENDEN
C
C               THE SCHEME OF STORAGE FOR AMPLTD(J) AND PHASQ(J) IS THE
C               SAME AS FOR OMMOD(J) AND VPMOD(J).  SEE SUBROUTINE ALLMO
C
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C
C AUTHORS     - A.D.PIERCE AND J.POSEY, M.I.T., JULY,1968
C
C                   ----CALLING SEQUENCE----
C
C SEE THE MAIN PROGRAM
C     DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),AMP(1)
C     DIMENSION AMPLTD(1),PHASQ(1)
C THESE QUANTITIES MUST BE DIMENSIONED.  THE PROGRAM USES VARIABLE
C DIMENSIONING.  FOR ACTUAL DIMENSIONS ASSIGNED, SEE THE MAIN PROGRAM.
C     CALL PPAMP(YIELD,MOFND,KST,KFIN,OMMOD,VPMOD,AMP,ALAM,FACT,
C    1 AMPLTD,PHASQ)
C
C                   ----EXTERNAL SUBROUTINES REQUIRED----
C
C     SOURCE, PHASE     (PHASE IS CALLED BY SOURCE)
C
C                   ----ARGUMENT LIST----
C
C     YIELD        R*4     NO      INP
C     MOFND        I*4     NO      INP
C     KST          I*4     VAR     INP
C     KFIN         I*4     VAR     INP
C     OMMOD        R*4     VAR     INP
C     VPMOD        R*4     VAR     INP
C     AMP          R*4     VAR     INP
C     ALAM         R*4     NO      INP
C     FACT         R*4     NO      INP
C     AMPLTD       R*4     VAR     OUT
C     PHASQ        R*4     VAR     OUT
C
C NO COMMON STORAGE IS USED
C
C                   ----INPUTS----
C
C     YIELD        =ENERGY RELEASE OF EXPLOSION IN EQUIVALENT KILOTONS O
C                   TNT, 1 KT = 4.2E19 ERGS.
C     MOFND        =NUMBER OF MODES FOUND IN PREVIOUS TABULATION OF
C                   DISPERSION CURVES.
C     KST(N)       =INDEX OF FIRST TABULATED POINT IN N-TH MODE.
C     KFIN(N)      =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN
C                   GENERAL, KFIN(N)=KST(N+1)-1.
C     OMMOD(N)     =ARRAY STORING ANGULAR FREQUENCY ORDINATE OF POINTS
C                   ON DISPERSION CURVES.  THE NMODE MODE IS STORED FOR
C                   N BETWEEN KST(NMODE) AND KFIN(NMODE).
C     VPMOD(N)     =ARRAY STORING PHASE VELOCITY ORDINATE OF POINTS ON
C                   DISPERSION CURVES.  THE NMODE MODE IS STORED FOR
C                   N BETWEEN KST(NMODE) AND KFIN(NMODE).
C     AMP(N)       =AMPLITUDE FACTOR INDEPENDENT OF YIELD COMPUTED BY
C                   SUBROUTINE PAMPDE CORRESPONDING TO ANGULAR FREQUENCY
C                   OMMOD(N) AND PHASE VELOCITY VPMOD(N).
C     ALAM         =A SCALING FACTOR DEPENDENT ON HEIGHT OF BURST, EQUAL
```

```
C                   TO CUBE ROOT OF (PRESSURE AT GROUND)/(PRESSURE AT    PPAMP
C                   BURST HEIGHT) TIMES (SOUND SPEED AT GROUND)/SOUND    PPAMP
C                   SPEED AT BURST HEIGHT).                              PPAMP
C     FACT         =A GENERAL AMPLITUDE FACTOR DEPENDENT ON BURST HEIGHT PPAMP
C                   AND OBSERVER HEIGHT.  A PRECISE DEFINITION IS GIVEN  PPAMP
C                   IN THE LISTING OF SUBROUTINE PAMPDE.                 PPAMP
C                                                                        PPAMP
C                   ----OUTPUTS----                                      PPAMP
C                                                                        PPAMP
C     AMPLTD(N)    =AMPLITUDE FACTOR REPRESENTING TOTAL MAGNITUDE OF     PPAMP
C                   FOURIER TRANSFORM OF THE CONTRIBUTION TO THE WAVEFOR PPAMP
C                   OF A SINGLE GUIDED MODE AT FREQUENCY OMMOD(N).  IT   PPAMP
C                   REPRESENTS THE AMPLITUDE OF THE NMODE-TH MODE IF N I PPAMP
C                   BETWEEN KST(NMODE) AND KFIN(NMODE), INCLUSIVE.  THE  PPAMP
C                   PRECISE DEFINITION IS GIVEN IN THE ABSTRACT.         PPAMP
C     PHASQ(N)     =PHASE LAG AT FREQUENCY OMMOD(N) FOR NMODE-TH MODE WH PPAMP
C                   N IS BETWEEN KST(NMODE) AND KFIN(NMODE), INCLUSIVE.  PPAMP
C                   THE INTEGRAND IS UNDERSTOOD TO HAVE THE FORM         PPAMP
C                   AMPLTD*COS(OMMOD*(TIME-DISTANCE/VPMOD)+PHASQ).       PPAMP
C                                                                        PPAMP
C                   ----PROGRAM FOLLOWS BELOW----                        PPAMP
C                                                                        PPAMP
C                                                                        PPAMP
C DIMENSION STATEMENTS USING VARIABLE DIMENSIONING                       PPAMP
      DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),AMP(1)                  PPAMP
      DIMENSION AMPLTD(1),PHASQ(1)                                       PPAMP
C                                                                        PPAMP
      Q=(YIELD)**(0.333333)                                              PPAMP
      ALAMP=Q*ALAM                                                       PPAMP
C                                                                        PPAMP
C START OF DO LOOP. II IS MODE NUMBER                                    PPAMP
      DO 20  II=1,MOFNO                                                  PPAMP
      K1=KST(II)                                                         PPAMP
      K2=KFIN(II)                                                        PPAMP
C                                                                        PPAMP
      DO 20  J=K1,K2                                                     PPAMP
C COMPUTATION OF SCALED FREQUENCY OMERAT                                 PPAMP
      OMERAT=OMMOD(J)*ALAMP                                              PPAMP
C COMPUTATION OF SQRT(K)                                                 PPAMP
      AKAYSQ = ABS(OMMOD(J)/VPMOD(J))                                    PPAMP
      AKAY = SQRT(AKAYSQ)                                                PPAMP
C                                                                        PPAMP
      CALL SOURCE(OMERAT,FTMAG,FTPHSE,DMAG,DPHSE)                        PPAMP
      AMPLTD(J)=(Q**2)*FACT*FTMAG*AMP(J)*AKAY/OMMOD(J)                   PPAMP
   20 PHASQ(J)=.75*3.14159-FTPHSE                                        PPAMP
C END OF DO LOOP                                                         PPAMP
C                                                                        PPAMP
      RETURN                                                             PPAMP
      END                                                                PPAMP
```

```
      SUBROUTINE PRATMO                                                 PRATMO
C     PRATMO (SUBROUTINE)                 9/1/68    LAST CARD IN DECK IS PRATMO
C                                                                       PRATMO
C                      ----ABSTRACT----                                 PRATMO
C                                                                       PRATMO
C TITLE - PRATMO                                                        PRATMO
C        PROGRAM TO PRINT OUT PARAMETERS DEFINING THE MODEL MULTILAYER  PRATMO
C        ATMOSPHERE.  A LISTING IS PRINTED OF LAYER NUMBER, HEIGHT OF   PRATMO
C        LAYER BOTTOM, HEIGHT OF LAYER TOP, LAYER THICKNESS, SOUND SPEE PRATMO
C        AND OF X AND Y COMPONENTS OF WIND VELOCITY.                    PRATMO
C                                                                       PRATMO
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            PRATMO
C                                                                       PRATMO
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., AUGUST,1968              PRATMO
C                                                                       PRATMO
C                      ----CALLING SEQUENCE----                         PRATMO
C                                                                       PRATMO
C SEE THE MAIN PROGRAM                                                  PRATMO
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                       PRATMO
C     COMMON IMAX,CI,VXI,VYI,HI  (THESE MUST BE IN COMMON)              PRATMO
C     CALL PRATMO                                                       PRATMO
C                                                                       PRATMO
C                      ----EXTERNAL SUBROUTINES REQUIRED----            PRATMO
C                                                                       PRATMO
C NO EXTERNAL SUBROUTINES ARE REQUIRED.                                 PRATMO
C                                                                       PRATMO
C                      ----ARGUMENT LIST----                            PRATMO
C                                                                       PRATMO
C COMMON STORAGE USED                                                   PRATMO
C     COMMON IMAX,CI,VXI,VYI,HI                                         PRATMO
C                                                                       PRATMO
C     IMAX        I*4    NO     INP                                     PRATMO
C     CI          R*4    100    INP                                     PRATMO
C     VXI         R*4    100    INP                                     PRATMO
C     VYI         R*4    100    INP                                     PRATMO
C     HI          R*4    100    INP                                     PRATMO
C                                                                       PRATMO
C                      ----INPUTS----                                   PRATMO
C                                                                       PRATMO
C     IMAX        =NUMBER OF LAYERS OF FINITE THICKNESS                 PRATMO
C     CI(I)       =SOUND SPEED IN KM/SEC IN I-TH LAYER                  PRATMO
C     VXI(I)      =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)  PRATMO
C     VYI(I)      =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)  PRATMO
C     HI(I)       =THICKNESS IN KM OF I-TH LAYER OF FINITE THICKNESS    PRATMO
C                                                                       PRATMO
C                      ----OUTPUTS----                                  PRATMO
C                                                                       PRATMO
C THE ONLY OUTPUT IS A PRINTOUT                                         PRATMO
C                                                                       PRATMO
C                      ----EXAMPLE----                                  PRATMO
C                                                                       PRATMO
C                MODEL ATMOSPHERE OF 10 LAYERS     (TOP OF NEW PAGE)    PRATMO
C                                                  (IMAX = 9)           PRATMO
C                                                                       PRATMO
C  LAYER         ZB          ZT          H           C          VX      PRATMO
C   10          22.50     INFINITE    INFINITE     0.2972     0.0082    PRATMO
C    9          20.00       22.50       2.50       0.2958     0.0093    PRATMO
C    8          17.50       20.00       2.50       0.2938     0.0118    PRATMO
C    7          15.00       17.50       2.50       0.2931     0.0144    PRATMO
C    6          12.50       15.00       2.50       0.2931     0.0165    PRATMO
C    5          10.00       12.50       2.50       0.2951     0.0160    PRATMO
C    4           7.50       10.00       2.50       0.3012     0.0149    PRATMO
C    3           5.00        7.50       2.50       0.3117     0.0118    PRATMO
C    2           2.50        5.00       2.50       0.3268     0.0098    PRATMO
```

```
C       1           0.          2.50         2.50       0.3394      0.0057      PRATMO
C                                                                                PRATMO
C                    ZB=HEIGHT OF LAYER BOTTOM IN KM                             PRATMO
C                    ZT=HEIGHT OF LAYER TOP IN KM           (THE VY COLUMN IS PRATMO
C                    H =WIDTH OF LAYER IN KM                 NOT SHOWN BECAUS PRATMO
C                    C =SOUND SPEED IN KM/SEC                OF LACK OF SPACE PRATMO
C                    VX=X COMP. OF WIND VEL. IN KM/SEC       IT DOES APPEAR O PRATMO
C                    VY=Y COMP. OF WIND VEL. IN KM/SEC       PRINTOUT.)       PRATMO
C                                                                              PRATMO
C                       ----PROGRAM FOLLOWS BELOW----                          PRATMO
C                                                                              PRATMO
C                                                                              PRATMO
C DIMENSION AND COMMON STATEMENTS LOCATING INPUT                              PRATMO
      DIMENSION CI(100),VXI(100),VYI(100),HI(100),ZI(100)                     PRATMO
      COMMON IMAX,CI,VXI,VYI,HI                                               PRATMO
C                                                                              PRATMO
C LET JET DENOTE THE INDEX OF THE UPPER HALFSPACE                             PRATMO
      JET=IMAX+1                                                              PRATMO
C                                                                              PRATMO
C PRINTING OF HEADING                                                         PRATMO
      WRITE (6,11) JET                                                        PRATMO
   11 FORMAT(1H1,14X,15HMODEL ATMOSPHERE OF,I4,7H LAYERS//)                   PRATMO
      WRITE (6,21)                                                            PRATMO
   21 FORMAT(1H ,2X,5HLAYER,7X,2HZB,10X,2HZT,11X,1HH,11X,1HC,11X,2HVX,        PRATMO
     110X,2HVY)                                                               PRATMO
C                                                                              PRATMO
      IF(IMAX .EQ. 0) GO TO 33                                                PRATMO
C                                                                              PRATMO
C ZI(I) DENOTES THE HEIGHT OF TOP OF I-TH LAYER OF FINITE THICKNESS          PRATMO
      ZI(1)=HI(1)                                                             PRATMO
      IF(IMAX .EQ. 1) GO TO 31                                                PRATMO
      DO 30 I=2,IMAX                                                          PRATMO
   30 ZI(I)=ZI(I-1)+HI(I)                                                     PRATMO
   31 CONTINUE                                                                PRATMO
C                                                                              PRATMO
C PRINTOUT FOR UPPER HALFSPACE                                                PRATMO
      XUV=ZI(IMAX)                                                            PRATMO
   33 IF(IMAX .EQ. 0) XUV=0.0                                                 PRATMO
      C=CI(JET)                                                               PRATMO
      VX=VXI(JET)                                                             PRATMO
      VY=VYI(JET)                                                             PRATMO
      WRITE (6,41) JET,XUV,C,VX,VY                                            PRATMO
   41 FORMAT(1H ,I7,F12.2,4X,8HINFINITE,4X,8HINFINITE,3F12.4)                 PRATMO
C                                                                              PRATMO
      IF(IMAX .EQ. 0) GO TO 60                                                PRATMO
      IF(IMAX .EQ. 1) GO TO 52                                                PRATMO
C                                                                              PRATMO
C TABULATION FOR LAYERS 2 THROUGH IMAX                                        PRATMO
      DO 50  J=2,IMAX                                                         PRATMO
      I=IMAX+2-J                                                              PRATMO
      IL=I-1                                                                  PRATMO
   50 WRITE (6,51) I,ZI(IL),ZI(I),HI(I),CI(I),VXI(I),VYI(I)                   PRATMO
   51 FORMAT(1H ,I7,3F12.2,3F12.4)                                            PRATMO
C                                                                              PRATMO
C TABULATION FOR LAYER 1                                                      PRATMO
   52 I=1                                                                     PRATMO
      USTED=0.0                                                               PRATMO
      WRITE (6,51) I,USTED,ZI(I),HI(I),CI(I),VXI(I),VYI(I)                    PRATMO
C                                                                              PRATMO
C PRINTOUT OF EXPLANATIONS                                                    PRATMO
   60 WRITE (6,61)                                                            PRATMO
   61 FORMAT(1H0,15X,31HZB=HEIGHT OF LAYER BOTTOM IN KM/ 1H ,15X,28HZT=       PRATMO
     1EIGHT OF LAYER TOP IN KM/1H ,15X,23HH =WIDTH OF LAYER IN KM/1H ,        PRATMO
     215X,24HC =SOUND SPEED IN KM/SEC/1H ,15X,33HVX=X COMP. OF WIND VEL       PRATMO
     3 IN KM/SEC/1H ,15X,33HVY=Y COMP. OF WIND VEL. IN KM/SEC)                PRATMO
C                                                                              PRATMO
      RETURN                                                                  PRATMO
      END                                                                     PRATMO
```

```
      SUBROUTINE RRRR(OMEGA,AKX,AKY,RPP,K)                              RRRR
C     RRRR (SUBROUTINE)                     8/1/68    LAST CARD IN DECK IS RRRR
C                                                                       RRRR
C                   ----ABSTRACT----                                    RRRR
C                                                                       RRRR
C TITLE - RRRR                                                          RRRR
C          THIS SUBROUTINE COMPUTES A 2-BY-2 TRANSFER MATRIX WHICH CONNEC RRRR
C          SOLUTIONS OF THE RESIDUAL EQUATIONS AT THE BOTTOM OF THE UPPER RRRR
C          HALFSPACE TO SOLUTIONS AT THE GROUND BY THE RELATIONS        RRRR
C                                                                       RRRR
C               PHI1(GROUND)= RPP(1,1)*PHI1(ZT(IMAX))+RPP(1,2)*PHI2(ZT(IM RRRR
C                                                                       RRRR
C               PHI2(GROUND)= RPP(2,1)*PHI1(ZT(IMAX))+RPP(2,2)*PHI2(ZT(IM RRRR
C                                                                       RRRR
C          WHERE ZT(IMAX) IS THE HEIGHT OF THE TOP OF THE IMAX LAYER AND  RRRR
C          CONSEQUENTLY THE HEIGHT OF THE BOTTOM OF THE UPPER HALFSPACE.  RRRR
C          THE FUNTIONS PHI1(Z) AND PHI2(Z) SATISFY THE RESIDUAL EQUATION RRRR
C                                                                       RRRR
C               D(PHI1)/DZ = A(1,1)*PHI1(Z) + A(1,2)*PHI2(Z)            RRRR
C                                                                       RRRR
C               D(PHI2)/DZ = A(2,1)*PHI1(Z) + A(2,2)*PHI2(Z)            RRRR
C                                                                       RRRR
C          WHERE THE A(I,J) ARE FUNCTIONS OF ALTITUDE BUT CONSTANT IN EAC RRRR
C          LAYER.                                                       RRRR
C                                                                       RRRR
C          IF WE LET EM(I) BE THE EM MATRIX (COMPUTED BY SUBROUTINE MMMM) RRRR
C          FOR THE I-TH LAYER, THEN (IN MATRIX NOTATION)                RRRR
C                                                                       RRRR
C             RPP = EM(1)*EM(2)*.....*EM(IMAX-1)*EM(IMAX)               RRRR
C                                                                       RRRR
C          THE ABOVE FORMULA IS USED TO COMPUTE THE RPP(I,J).           RRRR
C                                                                       RRRR
C          THE PARAMETERS DEFINING THE MULTILAYER ATMOSPHERE ARE PRESUMED RRRR
C          TO BE STORED IN COMMON.                                      RRRR
C                                                                       RRRR
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            RRRR
C                                                                       RRRR
C AUTHOR     - A.D.PIERCE, M.I.T., AUGUST,1968                          RRRR
C                                                                       RRRR
C                   ----CALLING SEQUENCE----                            RRRR
C                                                                       RRRR
C SEE SUBROUTINE NMFN                                                   RRRR
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100)                       RRRR
C     COMMON IMAX,CI,VXI,VYI,HI      (THESE MUST BE STORED IN COMMON    RRRR
C     DIMENSION RPP(2,2)                                                RRRR
C     CALL RRRR(OMEGA,AKX,AKY,RPP,K)                                    RRRR
C                                                                       RRRR
C                   ----EXTERNAL SUBROUTINES REQUIRED----               RRRR
C                                                                       RRRR
C     MMMM,AAAA,CAI,SAI                                                 RRRR
C                                                                       RRRR
C                   ----ARGUMENT LIST----                               RRRR
C                                                                       RRRR
C     OMEGA       R*4     NO      INP                                   RRRR
C     AKX         R*4     NO      INP                                   RRRR
C     AKY         R*4     NO      INP                                   RRRR
C     RPP         R*4   2-BY-2  OUT                                     RRRR
C     K           I*4     NO      OUT   (ALWAYS OUTPUT AS K=0)          RRRR
C                                                                       RRRR
C COMMON STORAGE USED                                                   RRRR
C     COMMON IMAX,CI,VXI,VYI,HI                                         RRRR
C                                                                       RRRR
C     IMAX        I*4     NO      INP                                   RRRR
C     CI          R*4     100     INP                                   RRRR
```

```
C     VXI          R*4     100    INP
C     VYI          R*4     100    INP
C     HI           R*4     100    INP
C
C                   ----INPUTS----
C
C     OMEGA        =ANGULAR FREQUENCY IN RAD/SEC
C     AKX          =X COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     AKY          =Y COMPONENT OF HORIZONTAL WAVE NUMBER VECTOR IN 1/KM
C     IMAX         =NUMBER OF LAYERS OF FINITE THICKNESS
C     CI(I)        =SOUND SPEED IN KM/SEC IN I-TH LAYER
C     VXI(I)       =X COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)
C     VYI(I)       =Y COMPONENT OF WIND VELOCITY IN I-TH LAYER (KM/SEC)
C     HI(I)        =THICKNESS IN KM OF I-TH LAYER OF FINITE THICKNESS
C
C                   ----OUTPUTS----
C
C     RPP          =2-BY-2 TRANSFER MATRIX WHICH CONNECTS SOLUTIONS OF
C                   THE RESIDUAL EQUATIONS AT THE BOTTOM OF THE UPPER
C                   HALFSPACE TO SOLUTIONS AT THE GROUND.
C     K            =DUMMY PARAMETER ALWAYS RETURNED AS 0.
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C DIMENSION AND COMMON STATEMENTS LOCATING PARAMETERS DEFINING THE MODE
C MULTILAYER ATMOSPHERE
      DIMENSION CI(100),VXI(100),VYI(100),HI(100)
      COMMON IMAX,CI,VXI,VYI,HI
C
      DIMENSION EM(2,2),AINT(2,2),RPP(2,2)
      K=0
C
C RPP AT TOP OF IMAX LAYER IS THE IDENTITY MATRIX
      RPP(1,1)=1.0
      RPP(1,2)=0.0
      RPP(2,1)=0.0
      RPP(2,2)=1.0
C
C START OF DO LOOP RUNNING THROUGH IMAX LAYERS IN DESCENDING ORDER
      DO 100 JASA=1,IMAX
      IASA=IMAX+1-JASA
C IASA IS THE INDEX OF THE LAYER CURRENTLY UNDER CONSIDERATION
C
C COMPUTATION OF EM MATRIX FOR IASA LAYER
      C=CI(IASA)
      VX=VXI(IASA)
      VY=VYI(IASA)
      H=HI(IASA)
      CALL MPPM(OMEGA,AKX,AKY,C,VX,VY,H,EM)
C
C MULTIPLICATION OF RPP AT TOP OF IASA LAYER BY EM FOR IASA LAYER
      DO 80 I=1,2
      DO 80 J=1,2
   80 AINT(I,J)=EM(I,1)*RPP(1,J)+EM(I,2)*RPP(2,J)
C
C CURRENT AINT IS RPP AT BOTTOM OF IASA LAYER
      DO 85 I=1,2
      DO 85 J=1,2
   85 RPP(I,J)=AINT(I,J)
C
  100 CONTINUE
C END OF OUTER DO LOOP
C
C CURRENT RPP IS THAT AT BOTTOM OF FIRST LAYER
      RETURN
      END
```

```
      SUBROUTINE RTMI(X,F,FCT,XLI,XRI,EPS,IEND,IER)
C
C     ..................................................................
C
C        SUBROUTINE RTMI
C
C        PURPOSE
C           TO SOLVE GENERAL NONLINEAR EQUATIONS OF THE FORM FCT(X)=0
C           BY MEANS OF MUELLER-S ITERATION METHOD.
C
C        USAGE
C           CALL RTMI (X,F,FCT,XLI,XRI,EPS,IEND,IER)
C           PARAMETER FCT REQUIRES AN EXTERNAL STATEMENT.
C
C        DESCRIPTION OF PARAMETERS
C           X      - RESULTANT ROOT OF EQUATION FCT(X)=0.
C           F      - RESULTANT FUNCTION VALUE AT ROOT X.
C           FCT    - NAME OF THE EXTERNAL FUNCTION SUBPROGRAM USED.
C           XLI    - INPUT VALUE WHICH SPECIFIES THE INITIAL LEFT BOUND
C                    OF THE ROOT X.
C           XRI    - INPUT VALUE WHICH SPECIFIES THE INITIAL RIGHT BOUN
C                    OF THE ROOT X.
C           EPS    - INPUT VALUE WHICH SPECIFIES THE UPPER BOUND OF THE
C                    ERROR OF RESULT X.
C           IEND   - MAXIMUM NUMBER OF ITERATION STEPS SPECIFIED.
C           IER    - RESULTANT ERROR PARAMETER CODED AS FOLLOWS
C                     IER=0 - NO ERROR,
C                     IER=1 - NO CONVERGENCE AFTER IEND ITERATION STEPS
C                             FOLLOWED BY IEND SUCCESSIVE STEPS OF
C                             BISECTION,
C                     IER=2 - BASIC ASSUMPTION FCT(XLI)*FCT(XRI) LESS
C                             THAN OR EQUAL TO ZERO IS NOT SATISFIED.
C
C        REMARKS
C           THE PROCEDURE ASSUMES THAT FUNCTION VALUES AT INITIAL
C           BOUNDS XLI AND XRI HAVE NOT THE SAME SIGN. IF THIS BASIC
C           ASSUMPTION IS NOT SATISFIED BY INPUT VALUES XLI AND XRI, TH
C           PROCEDURE IS BYPASSED AND GIVES THE ERROR MESSAGE IER=2.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           THE EXTERNAL FUNCTION SUBPROGRAM FCT(X) MUST BE FURNISHED
C           BY THE USER.
C
C        METHOD
C           SOLUTION OF EQUATION FCT(X)=0 IS DONE BY MEANS OF MUELLER-S
C           ITERATION METHOD OF SUCCESSIVE BISECTIONS AND INVERSE
C           PARABOLIC INTERPOLATION, WHICH STARTS AT THE INITIAL BOUNDS
C           XLI AND XRI. CONVERGENCE IS QUADRATIC IF THE DERIVATIVE OF
C           FCT(X) AT ROOT X IS NOT EQUAL TO ZERO. ONE ITERATION STEP
C           REQUIRES TWO EVALUATIONS OF FCT(X). FOR TEST ON SATISFACTOR
C           ACCURACY SEE FORMULAE (3,4) OF MATHEMATICAL DESCRIPTION.
C           FOR REFERENCE, SEE G. K. KRISTIANSEN, ZERO OF ARBITRARY
C           FUNCTION, BIT, VOL. 3 (1963), PP.205-206.
C
C     ..................................................................
C
C
C
C     PREPARE ITERATION
      IER=0
      XL=XLI
      XR=XRI
      X=XL
      TOL=X
```

```
      F=FCT(TOL)
      IF(F)1,16,1
    1 CONTINUE
      CONV=ABS(F)
      F=FCT(TOL)/CONV
      FL=F
      X=XR
      TOL=X
      F=FCT(TOL)/CONV
      IF(F)2,16,2
    2 FR=F
      IF(SIGN(1.,FL)+SIGN(1.,FR)) 25,3,25
C
C     BASIC ASSUMPTION FL*FR LESS THAN 0 IS SATISFIED.
C     GENERATE TOLERANCE FOR FUNCTION VALUES.
    3 I=0
      TOLF=EPS*(ABS(F)+1.0)*100.0
C
C
C     START ITERATION LOOP
    4 I=I+1
C
C     START BISECTION LOOP
      DO 13 K=1,IEND
      X=.5*(XL+XR)
      TOL=X
      F=FCT(TOL)/CONV
      IF(F)5,16,5
    5 IF(SIGN(1.,F)+SIGN(1.,FR)) 7,6,7
C
C     INTERCHANGE XL AND XR IN ORDER TO GET THE SAME SIGN IN F AND FR
    6 TOL=XL
      XL=XR
      XR=TOL
      TOL=FL
      FL=FR
      FR=TOL
    7 TOL=F-FL
      A=F*TOL
      A=A+A
      IF(ABS(A)-1.0E35) 71,71,130
   71 CONTINUE
      QUEST=FR*(FR-FL)
      IF(ABS(QUEST)-1.0E35) 72,72,130
   72 CONTINUE
      IF(A-FR*(FR-FL))8,9,9
    8 IF(I-IEND)17,17,9
    9 XR=X
      FR=F
C
C     TEST ON SATISFACTORY ACCURACY IN BISECTION LOOP
      TOL=EPS
      A=ABS(XR)
      IF(A-1.)11,11,10
   10 TOL=TOL*A
   11 IF(ABS(XR-XL)-TOL)12,12,13
   12 IF(ABS(FR-FL)-TOLF)14,14,13
   13 CONTINUE
C     END OF BISECTION LOOP
C
C     NO CONVERGENCE AFTER IEND ITERATION STEPS FOLLOWED BY IEND
C     SUCCESSIVE STEPS OF BISECTION OR STEADILY INCREASING FUNCTION
C     VALUES AT RIGHT BOUNDS. ERROR RETURN.
  130 CONTINUE
```

```
      IER=1
      F=CONV*F
      FR=CONV*FR
      FL=CONV*FR
      Q=2.*(XL-XR)/(XL+XR)
      QA=ABS(Q)
      IF (QA .LT. 1.0E-4) GO TO 16
      IER = 1
      WRITE(6,723) IER,X
      WRITE(6,724) F,XLI,XRI
      WRITE(6,725) FR,FL,XR,XL,QA
      GO TO 16
   14 IF(ABS(FR)-ABS(FL))16,16,15
   15 X=XL
      F=FL
      FL=CONV*FR
      FR=CONV*FR
      F=CONV*F
   16 RETURN
C
C     COMPUTATION OF ITERATED X-VALUE BY INVERSE PARABOLIC INTERPOLATIO
   17 A=FR-F
      DX=(X-XL)*FL*(1.+F*(A-TOL)/(A*(FR-FL)))/TOL
      XM=X
      FM=F
      X=XL-DX
      TOL=X
      F=FCT(TOL)/CONV
      IF(F)18,16,18
C
C     TEST ON SATISFACTORY ACCURACY IN ITERATION LOOP
   18 TOL=EPS
      A=ABS(X)
      IF(A-1.)20,20,19
   19 TOL=TOL*A
   20 IF(ABS(DX)-TOL)21,21,22
   21 IF(ABS(F)-TOLF)16,16,22
C
C     PREPARATION OF NEXT BISECTION LOOP
   22 IF(SIGN(1.,F)+SIGN(1.,FL)) 24,23,24
   23 XR=X
      FR=F
      GO TO 4
   24 XL=X
      FL=F
      XR=XM
      FR=FM
      GO TO 4
C     END OF ITERATION LOOP
C
C
C     ERROR RETURN IN CASE OF WRONG INPUT DATA
   25 IER=2
      FL=CONV*FL
      FR=CONV*FR
      F=CONV*F
      IER = 2
      WRITE(6,723) IER,X
      WRITE(6,724) F,XLI,XRI
      WRITE(6,725) FR,FL,XR,XL,QA
  723 FORMAT(1H ,4X,15HTROUBLE IN RTMI.3X,4HIER=,I6,3X,2HX=,G15.8)
  724 FORMAT(1H ,3X,2HF=,G15.8,3X,8HXLI.XRI=,2G15.8)
  725 FORMAT(1H ,3X,6HFR,FL=,2G15.8,3X,6HXR,XL=,2G15.8,3X,3HQA=,G15.8)
      RETURN
      END
```

```
      FUNCTION SAI(X)                                                    SAI
C     SAI (FUNCTION)                      7/25/68   LAST CARD IN DECK IS SAI
C                                                                        SAI
C                      ----ABSTRACT----                                  SAI
C                                                                        SAI
C TITLE - SAI                                                            SAI
C           PROGRAM TO EVALUATE FUNCTION SAI(X) FOR GIVEN VARIABLE X.    SAI
C           IF X IS NEGATIVE, SAI(X)=SIN(Y)/Y WITH Y=SQRT(-X).  IF X IS  SAI
C           POSITIVE, SAI(X)=SINH(Y)/Y WITH Y=SQRT(X).   THE FUNCTION IS SAI
C           ALSO REPRESENTABLE BY THE POWER SERIES                       SAI
C                                                                        SAI
C               SAI(X)= 1 + X/(3FACT) + X**2/(5FACT) + X**3/(7FACT) + ...SAI
C                                                                        SAI
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             SAI
C                                                                        SAI
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968                             SAI
C                                                                        SAI
C                      ----CALLING SEQUENCE----                          SAI
C                                                                        SAI
C     SAI(ANY R*4 ARGUMENT) MAY BE USED IN ARITHMETIC EXPRESSIONS        SAI
C                                                                        SAI
C                      ----EXTERNAL SUBROUTINES REQUIRED----             SAI
C                                                                        SAI
C     NO EXTERNAL SUBROUTINES ARE REQUIRED                               SAI
C                                                                        SAI
C                      ----ARGUMENT LIST----                             SAI
C                                                                        SAI
C     X          R*4     NO     INP                                      SAI
C     SAI        R*4     NO     OUT                                      SAI
C                                                                        SAI
C NO COMMON STORAGE IS USED                                              SAI
C                                                                        SAI
C                      ----PROGRAM FOLLOWS BELOW----                     SAI
C                                                                        SAI
C                                                                        SAI
    1 IF( ABS(X) .GT. 1.E-15 ) GO TO 9                                   SAI
C                                                                        SAI
C ABS(X) IS SO SMALL THAT SAI IS VIRTUALLY 1.0                           SAI
      SAI=1.0                                                            SAI
      RETURN                                                             SAI
C                                                                        SAI
C CONTINUING FROM 1                                                      SAI
    9 Y=SQRT(ABS(X))                                                     SAI
      IF(X) 10,10,11                                                     SAI
C                                                                        SAI
C X IS LESS THAN 0.                                                      SAI
   10 SAI=SIN(Y)/Y                                                       SAI
      RETURN                                                             SAI
C                                                                        SAI
C X IS POSITIVE.  SAI= SINH(Y)/Y.                                        SAI
   11 E=EXP(Y)                                                           SAI
      SAI=0.5*(E-1./E)/Y                                                 SAI
      RETURN                                                             SAI
      END                                                                SAI
```

```
      SUBROUTINE SOURCE(OMEGA,FTMAG,FTPHSE,DMAG,DPHSE)                  SOURCE
C     SOURCE (SUBROUTINE)                   8/15/68                     SOURCE
C                                                                       SOURCE
C                                                                       SOURCE
C                    ----ABSTRACT----                                   SOURCE
C                                                                       SOURCE
C TITLE - SOURCE                                                        SOURCE
C        EVALUATION OF FOURIER TRANSFORM OF NEAR FIELD ACOUSTIC RESPONS SOURCE
C        TO EXPLOSIVE SOURCE                                            SOURCE
C                                                                       SOURCE
C              SOURCE COMPUTES THE FOURIER TRANSFORM OF THE NEAR FIELD  SOURCE
C              PRESSURE AT 1 KM FROM A 1 KT EXPLOSION AT SEA LEVEL.  TH SOURCE
C              AMBIENT PRESSURE IS ASSUMED TO BE 1.E6 DYNES/CM**2 AND   SOURCE
C              THE TIME LAPSE FROM TIME ZERO IS NEGLECTED.  AN EMPIRICA SOURCE
C              FORMULA FOR THIS PRESSURE IS                             SOURCE
C                                                                       SOURCE
C                 F(T) = PAS * (1 - (T/TAS)) * EXP( -T/TAS )  , T .GT.  SOURCE
C                                                                       SOURCE
C                      = 0    ,  T .LT. 0                               SOURCE
C                                                                       SOURCE
C              WITH PAS = (34.45E+3) * (1.61) DYNES/CM**2               SOURCE
C              AND  TAS = 0.33 SEC  .                                   SOURCE
C                                                                       SOURCE
C              THEREFORE, ITS FOURIER TRANSFORM IS                      SOURCE
C                                                                       SOURCE
C                 FT(OMEGA)= -I * OMEGA * PAS / (1/TAS - I * OMEGA)**2  SOURCE
C                                                                       SOURCE
C              WHERE I = (-1)**0.5  .                                   SOURCE
C                                                                       SOURCE
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            SOURCE
C                                                                       SOURCE
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., AUGUST,1968              SOURCE
C                                                                       SOURCE
C                                                                       SOURCE
C                    ----USAGE----                                      SOURCE
C                                                                       SOURCE
C     SUBROUTINE PHASE IS CALLED                                        SOURCE
C                                                                       SOURCE
C FORTRAN USAGE                                                         SOURCE
C                                                                       SOURCE
C     CALL SOURCE(OMEGA,FTMAG,FTPHSE,DMAG,DPHSE)                        SOURCE
C                                                                       SOURCE
C INPUTS                                                                SOURCE
C                                                                       SOURCE
C    OMEGA     ANGULAR FREQUENCY (RADIANS/SEC)                          SOURCE
C      R*4                                                              SOURCE
C                                                                       SOURCE
C OUTPUTS                                                               SOURCE
C                                                                       SOURCE
C    FTMAG     MAGNITUDE OF FT(OMEGA) DEFINED ABOVE IN SUBROUTINE ABSTR SOURCE
C      R*4     ( (DYNES/CM**2) / (RAD/SEC) )                            SOURCE
C                                                                       SOURCE
C    FTPHSE    PHASE OF FT(OMEGA) DEFINED ABOVE IN SUBROUTINE ABSTRACT  SOURCE
C      R*4     (RADIANS)                                                SOURCE
C                                                                       SOURCE
C    DMAG      DERIVATIVE OF FTMAG WITH RESPECT TO OMEGA ( (DYNES/CM**2 SOURCE
C      R*4     / (RAD/SEC)**2 )                                         SOURCE
C                                                                       SOURCE
C    DPHSE     DERIVATIVE OF FTPHSE WITH RESPECT TO OMEGA (RAD / (RAD/  SOURCE
C      R*4     SEC) )                                                   SOURCE
C                                                                       SOURCE
C                                                                       SOURCE
C                    ----PROGRAM FOLLOWS BELOW----                      SOURCE
C                                                                       SOURCE
```

```
C
C     WE ASSUME INVERSE R DEPENDENCE
      PAS=(34.45E+3/1.0)*(1.61)
C PAS IS IN DYNES/CM**2
C     THIS IS THE PEAK OVERPRESSURE AT 1 KM
      TAS=0.33
C     TAS IS THE LENGTH OF THE POSITIVE PHASE
      OMO=1.0/TAS
      DENOM=OMEGA**2+OMO**2
      FTMAG=PAS*OMEGA/DENOM
      DMAG=PAS/DENOM-2.0*PAS*OMEGA**2/DENOM**2
      CALL PHASE(OMO,OMEGA,X,PHI)
C     PHI IS THE ARCTAN OF OMEGA/OMO
      FTPHSE=-3.1415927/2.0+2.0*PHI
      DPHSE=2.0*OMO/DENOM
C     THE DERIVATIVE OF THE ARCTAN IS 1./(1.+Y**2)
      RETURN
      END
```

```
      SUBROUTINE SUSPCT(N,M,NROW,INMODE,ISUS)                           SUSPCT
C     SUSPCT (SUBROUTINE)                7/19/68   LAST CARD IN DECK IS SUSPCT
C                                                                       SUSPCT
C                                                                       SUSPCT
C                    ----ABSTRACT----                                   SUSPCT
C                                                                       SUSPCT
C TITLE - SUSPCT                                                        SUSPCT
C        EVALUATION OF SUSPICION INDEX OF ELEMENT (N,M) OF MATRIX INMOD SUSPCT
C                                                                       SUSPCT
C              SUSPCT EVALUATES THE SUSPICION INDEX, ISUS, OF THE ELEME SUSPCT
C              IN ROW N, COLUMN M OF THE MATRIX INMODE ( (N,M) MUST BE  SUSPCT
C              AN INTERIOR ELEMENT).  THE NEIGHBORS OF (N,M) ARE DEFINE SUSPCT
C              TO BE THE EIGHT ELEMENTS WHICH FORM THE THREE BY THREE   SUSPCT
C              ELEMENT SQUARE WHICH HAS (N,M) AT ITS CENTER.  THEY ARE  SUSPCT
C              NUMBERED FROM ONE TO NINE BEGINNING IN THE UPPER LEFT AN SUSPCT
C              PROCEEDING CLOCKWISE (NO. 1 AND NO. 9 ARE SAME ELEMENT). SUSPCT
C              EACH ELEMENT OF MATRIX INMODE MUST HAVE ONE OF THREE     SUSPCT
C              VALUES, -1, 1, OR 5.  (N,M) IS NOT SUSPICIOUS AND ISUS > SUSPCT
C              0 IF ANY ONE OF THE FOLLOWING CONDITIONS HOLDS.          SUSPCT
C                                                                       SUSPCT
C                   1.  ELEMENT (N,M) > 5                               SUSPCT
C                   2.  ANY OF ITS NEIGHBORS > 5                        SUSPCT
C                   3.  NOWHERE IN THE 3X3 ARRAY OF (N,M) AND ITS NEIGH SUSPCT
C                       BORS DOES THERE APPEAR TO BE A DISPERSION CURVE SUSPCT
C                       WITH POSITIVE SLOPE                             SUSPCT
C                                                                       SUSPCT
C              OTHERWISE ISUS IS SET EQUAL TO THE NUMBER OF THE QUADRAN SUSPCT
C              IN WHICH THE POSITIVE SLOPE APPEARS.  THE QUADRANTS ARE  SUSPCT
C              NUMBERED BEGINNING IN THE UPPER LEFT AND PROCEEDING CLOCK SUSPCT
C              WISE.                                                    SUSPCT
C                                                                       SUSPCT
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            SUSPCT
C                                                                       SUSPCT
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                SUSPCT
C                                                                       SUSPCT
C                    ----USAGE----                                      SUSPCT
C                                                                       SUSPCT
C    NO FORTRAN SUBROUTINES ARE CALLED                                  SUSPCT
C                                                                       SUSPCT
C FORTRAN USAGE                                                         SUSPCT
C    CALL SUSPCT(N,M,NROW,INMODE,ISUS)                                  SUSPCT
C                                                                       SUSPCT
C INPUTS                                                                SUSPCT
C                                                                       SUSPCT
C    N         ROW NUMBER OF ELEMENT UNDER CONSIDERATION (MAY NOT BE    SUSPCT
C      I*4     FIRST OR LAST ROW)                                       SUSPCT
C                                                                       SUSPCT
C    M         COLUMN NUMBER OF ELEMENT UNDER CONSIDERAION (MAY NOT BE  SUSPCT
C      I*4     FIRST OR LAST COLUMN)                                    SUSPCT
C                                                                       SUSPCT
C    NROW      TOTAL NUMBER OF ROWS IN INMODE                           SUSPCT
C      I*4                                                              SUSPCT
C                                                                       SUSPCT
C    INMODE    MATRIX UNDER CONSIDERATION STORED IN VECTOR FORM, COLUMN SUSPCT
C      I*4(D)  AFTER COLUMN.  EACH ELEMENT MUST BE -1, 1, OR 5.         SUSPCT
C                                                                       SUSPCT
C OUTPUTS                                                               SUSPCT
C                                                                       SUSPCT
C    ISUS      SUSPICION INDEX OF ELEMENT (N,M).  SEE ABSTRACT ABOVE FO SUSPCT
C      I*4     DEFINITION.                                              SUSPCT
C                                                                       SUSPCT
C                                                                       SUSPCT
C                    ----EXAMPLES----                                   SUSPCT
C                                                                       SUSPCT
C CALLING PROGRAM                                                       SUSPCT
C                                                                       SUSPCT
C    DIMENSION INMODE(9)                                                SUSPCT
C    INMODE > -1, -1, 1, 1, -1, 1, 1, 1, -1                             SUSPCT
```

```
C     CALL SUSPCT(2,2,3,INMODE,ISUS)
C     WRITE (6,200) ISUS
C 200 FORMAT (10H EXAMPLE 1,6X, 6HISUS >,I2)
C     INMODE > -1, -1, 1, 1, -1, -1, 1, 1, 1
C     CALL SUSPCT(2,2,3,INMODE,ISUS)
C     WRITE (6,300) ISUS
C 300 FORMAT (10H EXAMPLE 2,6X, 6HISUS >,I2)
C     END
C
C TABLES OF INMODE
C
C          EXAMPLE 1       EXAMPLE 2
C
C          -++             -++
C          --+             --+
C          ++-             +-+
C
C PRINTOUT
C
C     EXAMPLE 1      ISUS = 3
C     EXAMPLE 2      ISUS = 0
C
C
C                  ----PROGRAM FOLLOWS BELOW----
C
C
C
C VARIABLE DIMENSIONING OF INMODE
      DIMENSION IPP(9),INMODE(1)
C
C ELEMENT (N,M) OF INMODE IS ICEN
      J16=(M-1)*NROW+N
      ICEN=INMODE(J16)
      ISUS= 0
C
C IF ICEN IS 5, IT IS NOT SUSPICIOUS AND ISUS = 0
      IF(ICEN .EQ. 5) RETURN
C
C IPP(N) IS NEIGHBOR NO. N (SEE ABSTRACT ABOVE FOR NUMBERING SCHEME)
      J17=(M-2)*NROW+(N-1)
      IPP(1)=INMODE(J17)
      J18=(M-1)*NROW+(N-1)
      IPP(2)=INMODE(J18)
      J19=(M-0)*NROW+(N-1)
      IPP(3)=INMODE(J19)
      J20=(M-0)*NROW+(N-0)
      IPP(4)=INMODE(J20)
      J21=(M-0)*NROW+(N+1)
      IPP(5)=INMODE(J21)
      J22=(M-1)*NROW+(N+1)
      IPP(6)=INMODE(J22)
      J23=(M-2)*NROW+(N+1)
      IPP(7)=INMODE(J23)
      J24=(M-2)*NROW+(N+0)
      IPP(8)=INMODE(J24)
      IPP(9)= IPP(1)
      NX = 0
      DO 10 I=1,9
      IF(IPP(I) .EQ. 5) NX=NX+1
   10 CONTINUE
C NX IS THE NUMBER OF NEIGHBORS WHICH EQUAL +5
C
C IF MORE THAN ONE NEIGHBOR IS EQUAL TO +5, THEN ISUS=0
      IF (NX .GT. 1) RETURN
```

```
C                                                                          SUSPCT
C IF NEIGHBOR 3 IS THE ONLY ONE EQUAL TO +5 AND EITHER NEIGHBOR 2 OR       SUSPCT
C NEIGHBOR 4 DOES NOT AGREE WITH ICEN, THEN ISUS=2                         SUSPCT
      ISUM = IABS( ICEN + IPP(2) + IPP(4))                                 SUSPCT
      IF (IPP(3).EQ.5 .AND. ISUM.NE.3) ISUS=2                              SUSPCT
      IF (NX.GT.0) RETURN                                                  SUSPCT
   30 DO 50 I=1,9                                                          SUSPCT
   50 IPP(I)=(IABS(IPP(I)+ICEN))/2                                         SUSPCT
C IPP(I) IS 1 IF NEIGHBOR I AGREES WITH ICEN, IT IS 0 IF THEY DISAGREE     SUSPCT
C (TO REACH THIS POINT, NEITHER ICEN NOR ANY OF ITS NEIGHBORS COULD BE     SUSPCT
C                                                                          SUSPCT
      ISUS = 1                                                             SUSPCT
      IF( IPP(1) .EQ. 0 .AND. IPP(2) .EQ. 1 .AND. IPP(8) .EQ. 1)           SUSPCT
     1 RETURN                                                              SUSPCT
      IF( IPP(8) .EQ. 0 .AND. IPP(2) .EQ. 0) RETURN                        SUSPCT
      ISUS = 2                                                             SUSPCT
      IF( IPP(2) .EQ. 0 .AND. IPP(3) .EQ. 1) RETURN                        SUSPCT
      IF( IPP(3) .EQ. 1 .AND. IPP(4) .EQ. 0) RETURN                        SUSPCT
      ISUS = 3                                                             SUSPCT
      IF( IPP(5) .EQ. 0 .AND. IPP(4) .EQ. 1 .AND. IPP(6) .EQ. 1)           SUSPCT
     1 RETURN                                                              SUSPCT
      IF( IPP(4) .EQ. 0 .AND. IPP(6) .EQ. 0) RETURN                        SUSPCT
      ISUS = 4                                                             SUSPCT
      IF( IPP(6) .EQ. 0 .AND. IPP(7) .EQ. 1) RETURN                        SUSPCT
      IF( IPP(7) .EQ. 1 .AND. IPP(8) .EQ. 0) RETURN                        SUSPCT
      ISUS = 0                                                             SUSPCT
      RETURN                                                               SUSPCT
      END                                                                  SUSPCT
```

```
      SUBROUTINE TABLE(OM1,OM2,V1,V2,NCM,NVP,THETK,OM,V,INMODE,NOPT)    TABLE
C     TABLE (SUBROUTINE)                   7/19/68   LAST CARD IN DECK IS TABLE
C                                                                       TABLE
C                                                                       TABLE
C                ----ABSTRACT                                           TABLE
C                                                                       TABLE
C TITLE - TABLE                                                         TABLE
C          GENERATION OF SUSPICIONLESS TABLE OF NORMAL MODE DISPERSION  TABLE
C          FUNCTION SIGNS                                               TABLE
C                                                                       TABLE
C                TABLE CALLS SUBROUTINE MPOUT TO CONSTRUCT THE MATRIX OF TABLE
C                NORMAL MODE DISPERSION FUNCTION SIGNS INMODE (STORED IN TABLE
C                VECTOR FORM COLUMN AFTER COLUMN) FOR REGION IN FREQUENCY TABLE
C                PHASE VELOCITY PLANE (OM1.LE.OMEGA.LE.OM2.AND.V1.LE.VP.L TABLE
C                .V2).  SUBROUTINE SUSPCT IS CALLED TO EVALUATE THE SUSPI TABLE
C                CION INDEX ,ISUS, OF EACH INTERIOR ELEMENT IN THE MATRIX TABLE
C                SCANNING FROM LEFT TO RIGHT, TOP TO BOTTOM. IF ISUS .NE. TABLE
C                0 , INMODE IS ALTERED AS FOLLOWS.                      TABLE
C                    ISUS=1 ROW ADDED ABOVE SUSPICIOUS ELEMENT AND COLUM TABLE
C                           ADDED TO ITS LEFT                           TABLE
C                        =2 COLUMN ADDED TO RIGHT OF SUSPICIOUS ELEMENT TABLE
C                           AND ROW ADDED ABOVE IT                      TABLE
C                        =3 ROW ADDED BELOW SUSPICIOUS ELEMENT AND COLUM TABLE
C                           ADDED TO ITS RIGHT                          TABLE
C                        =4 COLUMN ADDED TO LEFT OF SUSPICIOUS ELEMENT  TABLE
C                           AND ROW ADDED BELOW IT                      TABLE
C                HOWEVER, NEITHER THE NUMBER OF ROWS NVP NOR THE NUMBER O TABLE
C                COLUMNS NOM WILL BE INCREASED BEYOND 100.  IF ISUS CALLS TABLE
C                FOR AN ADDITIONAL ROW WHEN NVP = 100 , THE MESSAGE     TABLE
C                (NVP = 100          N = XX      M = XX) WILL BE PRINTED. TABLE
C                N IS ROW NO. OF SUSPICIOUS ELEMENT.  M IS COLUMN NO.  IF TABLE
C                ISUS CALLS FOR ADDITION OF A COLUMN WHEN NOM = 100, THE  TABLE
C                MESSAGE (NOM = 100          N = XX      M = XX) IS PRINT TABLE
C                WHEN INMODE HAS BEEN EXPANDED SCANNING IS RESUMED AT THE TABLE
C                ELEMENT IN NEW MATRIX WITH SAME ROW AND COLUMN NOS. AS  TABLE
C                THOSE OF SUSPICIOUS ELEMENT IN OLD MATRIX.  IF NOPT IS  TABLE
C                POSITIVE INMODE WILL BE PRINTED AS IT IS RETURNED FROM  TABLE
C                MPOUT AND IN ITS FINAL FORM.                           TABLE
C                                                                       TABLE
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL - C28-6515-4)         TABLE
C                                                                       TABLE
C AUTHOR      - J.W.POSEY, M.I.T., JUNE,1968                            TABLE
C                                                                       TABLE
C                                                                       TABLE
C                    ----USAGE----                                      TABLE
C                                                                       TABLE
C     SUBROUTINES MPOUT,SUSPCT,LNGTHN,WIDEN,NMOFN ARE CALLED IN TABLE.  TABLE
C                                                                       TABLE
C FORTRAN USAGE                                                         TABLE
C     CALL TABLE(OM1,OM2,V1,V2,NCM,NVP,THETK,OM,V,INMODE,NOPT)          TABLE
C                                                                       TABLE
C INPUTS                                                                TABLE
C                                                                       TABLE
C    OM1         MINIMUM VALUE OF FREQUENCY TO BE CONSIDERED.           TABLE
C       R*4                                                             TABLE
C    OM2         MAXIMUM VALUE OF FREQUENCY TO BE CONSIDERED            TABLE
C       R*4                                                             TABLE
C    V1          MINIMUM VALUE OF PHASE VELOCITY TO BE CONSIDERED       TABLE
C       R*4                                                             TABLE
C    V2          MAXIMUM VALUE OF PHASE VELOCITY TO BE CONSIDERED       TABLE
```

```
C       R*4
C     NOM       INITIAL NO. OF FREQUENCIES TO BE CONSIDERED
C       I*4
C     NVP       INITIAL NO. OF PHASE VELOCITIES TO BE CONSIDERED
C       I*4
C     THETK     PHASE VELOCITY DIRECTION (RADIANS)
C       R*4
C     NOPT      PRINT OUT OPTION.  IF NOPT = -1, NO PRINT.  IF NOPT = 1,
C       I*4     INMODE IS PRINTED IN ITS INITIAL FORM (GENERATED BY MPOU
C               AND IN ITS FINAL FORM.
C
C OUTPUTS
C
C     NOM       TOTAL NO. OF FREQUENCIES CONSIDERED
C       I*4
C     NVP       TOTAL NO. OF PHASE VELOCITIES CONSIDERED
C       I*4
C     OM        VECTOR WHOSE ELEMENTS ARE THE VALUES OF ANGULAR FREQUENC
C       R*4(D)  CORRESPONDING TO THE COLUMNS OF THE INMODE MATRIX
C
C     V         VECTOR WHOSE ELEMENTS ARE THE VALUES OF PHASE VELOCITY
C       R*4(D)  CORRESPONDING TO THE ROWS OF THE INMODE MATRIX
C
C     INMODE    EACH ELEMENT OF THIS MATRIX CORRESPONDS TO A POINT IN TH
C       I*4(D)  FREQUENCY (OM) - PHASE VELOCITY (V) PLANE.  IF THE NORMA
C               MODE DISPERSION FUNCTION (FPP) IS POSITIVE AT THAT POINT
C               THE ELEMENT IS +1, IF FPP IS NEGATIVE, THE ELEMENT IS -1
C               IF FPP DOES NOT EXIST, THE ELEMENT IS 5.  INMODE HAS NVP
C               ROWS AND NOM COLUMNS.  MATRIX IS STORED AS A VECTOR,
C               COLUMN AFTER COLUMN.
C
C
C                     ----EXAMPLE----
C
C LET INMODE = -1,5,5,5,1,-1,-1,-1,1,1,-1,-1,1,1,1,1
C    WITH NOM = NVP = 4
C    AND OM = 1.0,1.5,2.0,2.5            THETK = 3.14159
C         V = 1.0,2.0,3.0,4.0
C (VALUES NOT CORRECT, FOR ILLUSTRATION ONLY)
C
C THEN THE TABLE WILL BE PRINTED AS FOLLOWS.
C
C VPHASE      NORMAL MODE DISPERSION FUNCTION SIGN
C  1.00000    -+++
C  2.00000    X-++
C  3.00000    X--+
C  4.00000    X--+
C
C      OMEGA 1234
C            PHASE VELOCITY DIRECTION IS    90.0000EGREES
C
C OMEGA =
C    0.10000E 01   0.15000E 01   0.20000E 01   0.25000E 01
C
C
C                   ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION OM(100),V(100),INMODE(10000),DORN(100),KORN(100)
      COMMON IMAX,CI(100),VXI(100),VYI(100),HI(100)
C
C MPOUT IS CALLED TO PRODUCE INMODE MATRIX AND OM AND V VECTORS.
      CALL MPOUT(OM1,OM2,V1,V2,NOM,NVP,INMODE,OM,V,THETK)
```

```
C
C IFLAG = 1 INDICATES FIRST TIME THROUGH WRITE PROCEDURE
      IFLAG = 1
C
C INMODE IS PRINTED IF NOPT IS POSITIVE
      IF (NOPT.GE.0) GO TO 123
    5 IFLAG = 0
      NOPER=0
C NOPER IS THE NUMBER OF EXPANSION OPERATIONS PERFORMED IN THE PRESENT
C SCAN OF THE MATRIX.  THUS, NOPER IS THE NUMBER OF SUSPICIOUS POINTS
C FOUND IN THE PRESENT SCAN.
C
C BEGIN SCANNING OF INTERIOR ELEMENTS OF INMODE IN UPPER LEFT CORNER
      N = 2
      M = 2
   10 CALL SUSPCT(N,M,NVP,INMODE,ISUS)
C
C POINT (N,M) IS SUSPICIOUS IF ISUS.NE.0
      IF(ISUS.NE.0) GO TO 60
C
C CHECK FOR END OF ROW
   20 IF (M.LT.(NOM-1)) GO TO 30
C
C CHECK FOR LAST ROW
      IF (N.LT.(NVP-1)) GO TO 40
      GO TO 121
C
C MOVE ONE COLUMN TO RIGHT
   30 M = M+1
      GO TO 10
C
C ADVANCE ONE ROW AND START AT COLUMN TWO
   40 N = N+1
      M = 2
      GO TO 10
C
C CHECK FOR MAXIMUM VALUE OF NVP
   60 IF(NVP.LT.100) GO TO 62
   61 FORMAT (24H NVP = 100             N =,I3,8H     M =,I3)
      WRITE (6,61) N,M
      GO TO 20
   62 IF(NOM .LT. 100) GO TO 70
   63 FORMAT(24HNOM = 100               N=,I3, 8H     M=,I3)
   64 WRITE(6,63) N,M
      GO TO 20
   70 IF(ISUS .NE. 1) GO TO 75
C
C  ADD ROW ABOVE SUSPICIOUS POINT
      N1=N-1
C
C  ADD A COLUMN TO LEFT OF SUSPICIOUS POINT
      M1=M-1
      GO TO 100
   75 IF(ISUS .NE. 2) GO TO 80
C
C  ADD A COLUMN TO RIGHT OF SUSPICIOUS POINT
      M1=M
C
C  ADD ROW ABOVE SUSPICIOUS POINT
      N1=N-1
      GO TO 100
   80 IF(ISUS .NE. 3) GO TO 85
C
C  ADD A COLUMN TO RIGHT OF SUSPICIOUS POINT
```

```
      M1=M
C
C  ADD ROW BELOW SUSPICIOUS POINT
      N1=N
      GO TO 100
C
C  ADD ROW BELOW SUSPICIOUS POINT
   85 N1=N
C
C  ADD A COLUMN TO LEFT OF SUSPICIOUS POINT
      M1=M-1
  100 CONTINUE
      CALL LNGTHN(OM,V,INMODE,NOM,NVP,NVPP,N1,1,THETK)
      CALL WIDEN(OM,V,INMODE,NOM,NOMP,NVPP,M1,1,THETK)
      NVP=NVPP
      NOM=NOMP
      NOPER=NOPER+1
      GO TO 10
  121 CONTINUE
      IF(NOPER .GT. 0 .AND. NVP .LT. 100 .AND. NOM .LT. 100) GO TO 5
C
C DO NOT PRINT INMODE IF NOPT IS NEGATIVE
      IF(NOPT .LT. 0) RETURN
C
C LABELING
  122 FORMAT (6H1VPHSE,6X,36HNORMAL MODE DISPERSION FUNCTION SIGN/)
  123 WRITE (6,122)
      DO 133 I=1,NVP
      DO 128 J=1,NOM
      J88=(J-1)*NVP+I
      J89=INMODE(J88)-1
      IF (J89) 126,125,124
  124 CONTINUE
C
C IF INMODE = 5, DORN = 1HX
      DATA Q1/1HX/
      DORN(J) = Q1
      GO TO 127
  125 CONTINUE
C
C IF INMODE = 1, DORN = 1H+
      DATA Q2/1H+/
      DORN(J) = Q2
      GO TO 127
  126 CONTINUE
C
C IF INMODE = -1, DORN = 1H-
      DATA Q3/1H-/
      DORN(J) = Q3
  127 CONTINUE
  128 CONTINUE
C
C PRINT ROW I OF TABLE
      WRITE (6,130)V(I),(DORN(J), J=1,NOM)
  130 FORMAT(1H ,F8.5,3X,100A1)
  133 CONTINUE
      J10 = 10
      DO 150 J=1,NOM
C
C NUMBER COLUMNS
  150 KORN(J) = MOD(J,J10)
      WRITE (6,213) (KORN(J), J=1,NOM)
  213 FORMAT (6HOOMEGA,6X,100I1)
C
```

```
C CONVERT THETK FROM RADIANS TO DEGREES
      X = THETK*180/3.14159
      WRITE (6,413) X
  413 FORMAT (1H ,11X,27HPHASE VELOCITY DIRECTION IS,F9.3,
     1   8HDEGREES )
      WRITE (6,513)
  513 FORMAT ( 8H0OMEGA =)
C
C LIST VALUES OF OMEGA WHICH CORRESPOND TO COLUMNS OF TABLE
      WRITE (6,613) (OM(I),I=1,NOM)
  613 FORMAT ( 1H ,5E14.5)
C
C IF SUSPICION ELIMINATION HAS NOT BEEN PERFORMED, BEGIN IT AT THIS TIM
      IF(IFLAG.EQ.1) GO TO 5
      RETURN
      END
```

```
      SUBROUTINE TABPRT(YIELD,MOFNO,KST,KFIN,OMMOD,VPMOD,               TABPRT
     1AMPLTO,PHASQ)                                                     TABPRT
C     TABPRT (SUBROUTINE)               7/31/68   LAST CARD IN DECK IS TABPRT
C                                                                       TABPRT
C                     ----ABSTRACT----                                  TABPRT
C                                                                       TABPRT
C TITLE - TABPRT                                                        TABPRT
C          PROGRAM TO PRINT OUT LISTS OF FREQUENCY, PHASE VELOCITY,     TABPRT
C          AMPLITUDE, AND PHASE FOR EACH GUIDED MODE EXCITED BY A NUCLEAR TABPRT
C          EXPLOSION OF GIVEN YIELD.  THE SIMULTANEOUS LISTING OF FREQUEN TABPRT
C          AND PHASE VELOCITY REPRESENTS THE DISPERSION CURVE FOR THE   TABPRT
C          GUIDED MODE.  THE QUANTITIES AMPLTO AND PHASE DEPEND ON SOURCE TABPRT
C          AND OBSERVER HEIGHTS AS WELL AS THE MODEL ATMOSPHERE.  HOWEVER TABPRT
C          THE LATTER INFORMATION IS NOT LISTED BY TABPRT AND IS PRESUMED TABPRT
C          TO BE LISTED BY ANOTHER SUBROUTINE.  THE SUBROUTINE TABPRT   TABPRT
C          SHOULD NOT BE CALLED UNTIL ALL THE QUANTITIES TO BE LISTED   TABPRT
C          HAVE BEEN COMPUTED AND STORED IN THE MACHINE.  NORMALLY,     TABPRT
C          ATMOS, TABLE, ALLMOD, PAMPDE, AND PPAMP WOULD BE CALLED BEFORE TABPRT
C          TABPRT.                                                      TABPRT
C                                                                       TABPRT
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            TABPRT
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., JULY,1968                TABPRT
C                                                                       TABPRT
C                     ----CALLING SEQUENCE----                          TABPRT
C                                                                       TABPRT
C     DIMENSION KST(1),KFIN(1),OMMOD(1),VPMOD(1),AMPLTO(1),PHASQ(1)     TABPRT
C THE SUBROUTINE USES VARIABLE DIMENSIONING.  THE TRUE DIMENSIONS MUST  TABPRT
C BE GIVEN IN THE PROGRAM WHICH DEFINES THESE QUANTITIES.  SEE THE      TABPRT
C DIMENSION STATEMENTS IN THE MAIN PROGRAM.                             TABPRT
C     CALL TABPRT(YIELD,MOFNO,KST,KFIN,OMMOD,VPMOD,AMPLTO,PHASQ)        TABPRT
C                                                                       TABPRT
C NO EXTERNAL SUBROUTINES ARE REQUIRED                                  TABPRT
C                                                                       TABPRT
C                     ----ARGUMENT LIST----                             TABPRT
C                                                                       TABPRT
C     YIELD        R*4     NO      INP                                  TABPRT
C     MOFNO        I*4     NO      INP                                  TABPRT
C     KST          I*4     VAR     INP                                  TABPRT
C     KFIN         I*4     VAR     INP                                  TABPRT
C     OMMOD        R*4     VAR     INP                                  TABPRT
C     VPMOD        R*4     VAR     INP                                  TABPRT
C     AMPLTO       R*4     VAR     INP                                  TABPRT
C     PHASQ        R*4     VAR     INP                                  TABPRT
C                                                                       TABPRT
C NO COMMON STORAGE USED                                                TABPRT
C                                                                       TABPRT
C                     ----INPUTS----                                    TABPRT
C                                                                       TABPRT
C     YIELD          =ENERGY YIELD OF EXPLOSION IN EQUIVALENT KILOTONS (KT TABPRT
C                     OF TNT.  1 KT = 4.2X(10)**19 ERGS.                TABPRT
C     MOFNO          =NUMBER OF NORMAL MODES FOUND                      TABPRT
C     KST(N)         =INDEX OF FIRST TABULATED POINT IN N-TH MODE       TABPRT
C     KFIN(N)        =INDEX OF LAST TABULATED POINT IN N-TH MODE.  IN   TABPRT
C                     GENERAL, KFIN(N)=KST(N+1)-1.                      TABPRT
C     OMMOD(N)       =ARRAY STORING ANGULAR FREQUENCY ORDINATE (RAD/SEC) O TABPRT
C                     POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR TABPRT
C                     FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).         TABPRT
C     VPMOD(N)       =ARRAY STORING PHASE VELOCITY ORDINATE (KM/SEC) OF TABPRT
C                     POINTS ON DISPERSION CURVES.  THE NMODE MODE IS STOR TABPRT
C                     FOR N BETWEEN KST(NMODE) AND KFIN(NMODE).         TABPRT
C     AMPLTO(N)      =AMPLITUDE FACTOR REPRESENTING TOTAL MAGNITUDE OF  TABPRT
C                     FOURIER TRANSFORM OF THE CONTRIBUTION TO THE WAVEFOR TABPRT
C                     FROM A SINGLE GUIDED MODE AT FREQUENCY OMMOD(N).  TABPRT
C                     ITS UNITS SHOULD BE (DYNES/CM**2)* (KM**(1/2))*SEC. TABPRT
```

```
C                    IT REPRESENTS THE AMPLITUDE OF NMODE-TH MODE IF N IS
C                    BETWEEN KST(NMODE) AND KFIN(NMODE), INCLUSIVE.  FOR
C                    PRECISE DEFINITION, SEE SUBROUTINE PPAMP.
C     PHASQ(N)      =PHASE LAG IN RADIANS AT FREQUENCY OMMCO(N) FOR THE
C                    NMODE-TH MODE WHEN N IS BETWEEN KST(NMODE) AND
C                    KFIN(NMODE), INCLUSIVE.  THE INTEGRAND IS UNDERSTOOD
C                    TO HAVE THE FORM AMPLTO*COS(OMMOD*(TIME-DISTANCE/VPM
C                    +PHASQ).  FOR A PRECISE DEFINITION, SEE SUBROUTINE
C                    PPAMP.
C
C                    ----OUTPUTS----
C
C PRINTOUT.  THE ONLY FUNCTION OF TABPRT IS TO PRINT OUT RESULTS.
C
C                    ----EXAMPLE----
C
C THE OUTPUT FORMAT IS ILLUSTRATED BELOW.
C
C                                MODE TABULATION FOR Y=   100.00 KILOTONS
C
C
C
C     MODE   1
C
C          OMEGA          VPHSE          AMPLTO          PHASE
C
C            .00100       0.33426  -7.01342E 20        -3.72139
C            .00200       0.24372  -8.02394E 20        -4.56028
C
C
C
C     MODE   2
C
C          OMEGA          VPHSE          AMPLTO          PHASE
C
C            .00100       0.55298  -7.95321E 10        -2.40798
C            .00200       0.48321  -1.23108E 11         2.30524
C
C ETC.
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C VARIABLE DIMENSIONING IS USED
      DIMENSION KST(1),KFIN(1),OMMCO(1),VPMOD(1),AMPLTO(1),PHASQ(1)
C
      WRITE (6,11) YIELD
   11 FORMAT( 1H1    ,1H ,25X,22HMODE TABULATION FOR Y=,F9.2,9H KILOTONS
C
C START OF OUTER DO LOOP
      DO 50  II=1,MOFNO
C
      WRITE (6,21) II
   21 FORMAT(1H ///,1H ,4X, 5HMODE ,I3//, 1H ,9X,5HOMEGA,9X,5HVPHSE,9X,
     1 6HAMPLTO,8X,5HPHASE/ )
C
      K1=KST(II)
      K2=KFIN(II)
C
C START OF INNER DO LOOP
      DO 50  J=K1,K2
C
   50 WRITE (6,51) OMMCO(J),VPMOD(J),AMPLTO(J),PHASQ(J)
   51 FORMAT( 1H ,4X,F14.5,F14.5,1PG14.5,0PF14.5)
C END OF LOOPS
C
      RETURN
      END
```

```
      SUBROUTINE THPT(TFIRST,TEND,DELTT,ROBS,
     1MOFNO,KST,KFIN,CMMOD,VPMOD,AKIS,AMPLTD,PHASQ,IOPT)
C       THPT (SUBROUTINE)                         7/19/68
C
C
C                    ----ABSTRACT----
C
C TITLE - THPT
C          CALCULATION AND PLOTTING OF FAR-FIELD TRANSIENT RESPONSE TO A
C          PRESSURE SOURCE IN THE ATMOSPHERE
C
C                THE RESPONSE OF MODE N IS FOUND BY INTEGRATING (AMPLTD *
C                COS( OMEGA * (T - R/VP) < PHASQ) OVER OMEGA FROM OMMOD
C                (KST(N)) TO CMMCO(KFIN(N)) AND DIVIDING BY SQRT(R). VP,
C                PHASQ, AND AMPLTD ARE FUNCTIONS OF BOTH N AND OMEGA.  TH
C                TOTAL RESPONSE IS THE SUM OF THE MODAL RESPONSES.  THE
C                RESPONSE IS CALCULATED FOR TIME TFIRST AND AT INTERVALS
C                OF DELTT THEREAFTER UNTIL TEND IS REACHED.  THE VALUE OF
C                IOPT DETERMINES WHAT WILL BE CALCULATED, PRINTED AND
C                PLOTTED.  (SEE INPUT LIST FOR POSSIBLE IOPT VALUES.) THE
C                RESULTS ARE TABULATED IN THE PRINTOUT AND GRAPHED BY THE
C                CALCOMP PLOTTER.
C
C                THE CURRENT VERSION OF THIS SUBROUTINE DIFFERS FROM THAT
C                REPORTED IN AFCRL-70-0134 IN THAT THE RESULTING THEORETI
C                CAL PRESSURE PERTURBATIONS INCLUDE THE EARTH CURVATURE
C                CORRECTION FACTOR, ( ROBS / ( RE * SIN(ROBS/RE) ) )**0.5
C
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C AUTHOR      - J.W.POSEY, M.I.T., JUNE,1968
C
C
C                    ----USAGE----
C
C
C       FORTRAN SUBROUTINE AKI IS CALLED
C
C       CALCOMP PLOTTER SUBROUTINES PLOT1, AXIS1, NUMBR1, SYMBLS, AND
C       SCLGPH ARE CALLED TO WRITE THE CALCOMP TAPE.  SUBROUTINE NEWPLT
C       MUST HAVE BEEN CALLED PRIOR TO CALLING THPT, AND ENDPLT MUST BE
C       CALLED AFTER RETURNING  FROM THPT.  (SEE MAIN PROGRAM)
C
C FORTRAN USAGE
C       CALL THPT(TFIRST,TEND,CELTT,ROBS,MOFNO,KST,KFIN,CMMOD,VPMOD,AMPLT
C      1  ,PHASQ,IOPT)
C
C INPUTS
C
C     TFIRST     TIME AT WHICH TABULATION AND PLOTTING OF RESPONSE IS TO
C       R*4      BEGIN  (SEC)
C
C     TEND       TIME AT WHICH TABULATION AND PLOTTING OF RESPONSE IS TO
C       R*4      END  (.LE.(TFIRST+5400.))  (SEC)
C
C     DELTT      TIME INTERVAL BETWEEN SUCCESSIVE CALCULATIONS OF THE
C       R*4      RESPONSE (.GE.((TEND-TFIRST)/1000))  (SEC)
C
C     ROBS       DISTANCE OF THE OBSERVOR FROM THE SOURCE OF THE DISTUR-
C       R*4      BANCE  (KM)
C
C     MOFNO      NUMBER OF MODES FOUND (.LE.10)
C       I*4
C
C     KST        ELEMENT N OF THIS VECTOR IS NUMBER OF OMMCO ELEMENT WHIC
C       I*4(D)   IS FIRST FREQUENCY CONSIDERED FOR MODE N
```

```
C                                                                       THMPT
C     KFIN        ELEMENT N OF THIS VECTOR IS NUMBER OF OMMOD ELEMENT WHIC THMPT
C       I*4(D)    IS LAST FREQUENCY CONSIDERED FOR MODE N               THMPT
C                                                                       THMPT
C     OMMOD       ELEMENTS OF THIS VECTOR NUMBERED KST(N) THROUGH KFIN(N)  THMPT
C       R*4(D)    ARE THE VALUES OF FREQUENCY (IN INCREASING ORDER) FOR THMPT
C                 WHICH THE CORRESPONDING MODE N PHASE VELOCITIES HAVE BEE THMPT
C                 DETERMINED                                            THMPT
C                                                                       THMPT
C     VPMOD       VECTOR OF PHASE VELOCITIES WHICH CORRESPOND TO THE FRE-  THMPT
C       R*4(D)    QUENCIES OF VECTOR OMMOD                              THMPT
C                                                                       THMPT
C     AMPLTD      VALUES OF AMPLITUDE FUNCTION IN AKI INTEGRAL (ELEMENTS   THMPT
C       R*4(D)    CORRESPOND DIRECTLY TO ELEMENTS OF OMMOD) (DYNES/CM**2)  THMPT
C                                                                       THMPT
C     PHASQ       TERM IN ARGUMENT OF COS IN AKI INTEGRAL WHICH IS INDEPEN THMPT
C       R*4(D)    DENT OF TIME AND DISTANCE (RO3S)                      THMPT
C                                                                       THMPT
C     IOPT        COMPUTATION AND PRINT OPTION INDICATOR                THMPT
C       I*4       = 1,2,...,10  CALCULATE, PRINT AND PLOT MODE NO. IOPT ON THMPT
C                 = 11  CALCULATE, PRINT AND PLOT ALL MODES AS WELL AS THE THMPT
C                      TOTAL RESPONSE                                   THMPT
C                 = 12  CALCULATE ALL MODES, PRINT AND PLOT TOTAL RESPONSE THMPT
C                      ONLY                                             THMPT
C                                                                       THMPT
C OUTPUTS                                                               THMPT
C                                                                       THMPT
C     THE ONLY OUTPUTS ARE THE PRINTOUTS AND PLOTS CALLED FOR BY IOPT.  THMPT
C     ALL GRAPHS ARE DRAWN TO THE SAME SCALE.  THE PRESSURE SCALE IS    THMPT
C     DETERMINED BY THE MAXIMUM AMPLITUDE OF THE TOTAL PRESSURE, AND THE THMPT
C     TIME SCALE IS 600 SEC PER INCH.  PRESSURE IS EXPRESSED IN DYNES/CM THMPT
C                                                                       THMPT
C                                                                       THMPT
C                      ----PROGRAM FOLLOWS BELOW----                    THMPT
C                                                                       THMPT
C                                                                       THMPT
C                                                                       THMPT
C                                                                       THMPT
      DIMENSION KST(10),KFIN(10),OMMOD(1000),VPMOD(1000),AMPLTD(1000),  THMPT
     1  PHASQ(1000),T(1001),TOTINT(1001),TNINT(10,1001),Y(1001)         THMPT
      DIMENSION CKI(1000)                                               THMPT
      DIMENSION AKIS(1000)                                              THMPT
C                                                                       THMPT
C YAX IS VECTOR OF LITERAL CONSTANTS.  ELEMENT N IS THE EIGHT SPACE LAB THMPT
C FOR THE PRESSURE AXIS ON THE GRAPH OF THE MODE N RESPONSE.            THMPT
      DOUBLE PRECISION YAX(10)                                          THMPT
      DATA YAX/8H MODE 1 ,8H MODE 2 ,8H MODE 3 ,8H MODE 4 ,8H MODE 5 ,  THMPT
     1  8H MODE 6 ,8H MODE 7 ,8H MODE 8 ,8H MODE 9 ,8H MODE 10/         THMPT
      IF(IOPT .NE. 11)  GO TO 4                                         THMPT
      WRITE (6,2)                                                       THMPT
    2 FORMAT (1H1, 40X,23HTABULATION OF RESPONSES//)                    THMPT
      WRITE (6,3)                                                       THMPT
    3 FORMAT (1H ,20X,4HTIME,12X,5HTOTAL,11X,6HMODE 1,10X,6HMODE 2,10X, THMPT
     1  6HMODE 3,10X,6HMODE 4,10X,6HMODE 5/)                            THMPT
    4 IF(IOPT.EQ.12) WRITE(6,753)                                       THMPT
  753 FORMAT (1H1,45X,40HTABULATION OF ACOUSTIC PRESSURE RESPONSE///1H  THMPT
     1  48X,10HTIME (SEC),9X,15HP (DYNES/CM**2)//)                      THMPT
C                                                                       THMPT
C L IS NUMBER OF TIMES AT  WHICH RESPONSE IS TO BE CALCULATED           THMPT
      L = (TEND -TFIRST) / DELTT + 1                                    THMPT
      L=MIN0(L,999)                                                     THMPT
C                                                                       THMPT
C SIZE IS THE LENGTH OF THE TIME AXIS IN INCHES                         THMPT
      SIZE = ( TEND - TFIRST ) / 600.0                                  THMPT
```

```
C
C PRESET ALL RESPONSE VALUES TO 0.0
    5 DO 7 K=1,L
      TOTINT(K) = 0.0
      DO 7  N=1,10
    7 TNINT(N,K) = 0.0
C
C SET UP TABLE OF TIMES BEGINNING AT TFIRST AND TAKING VALUES OF TIME A
C INTERVALS OF DELTT UNTIL TEND IS REACHED
    9 DO 10 IT=1,L
   10 T(IT) = TFIRST + (IT-1)*DELTT
C
C BEGIN SET UP TO CALCULATE MODE 1 RESPONSE
      N = 1
C
C IF IOPT.LE.10 CALCULATE ONLY MODE IOPT RESPONSE
      IF (IOPT.LE.10) N = IOPT
   11 NOST = KST(N) + 1
      NOFN = KFIN(N)
C
C DETERMINE THE EARTH CURVATURE CORRECTION FACTOR TIMES ROBS**(-0.5).
      RAD = ROBS / 6374.
      CF = (1./(6374.*ABS(SIN(RAD))))**0.5
C
C THE MODE N RESPONSE IS FOUND FOR ALL VALUES OF T BEFORE NEXT MODE IS
C CONSIDERED
      DO 51 IT=1,L
C
C SET A2,PH2 EQUAL TO VALUES FOR A1,PH1 IN FIRST INTEGRATION INTERVAL
      J26 = KST(N)
      A2 = AMPLTD(J26)
      A2 = A2*EXP(-AKIS(J26)*ROBS)
      NHWR = RAD/3.1415926535
      PHASQ(J26) = PHASQ(J26) + NHWR*(3.1415926535/2.0)
      S2=OMMOD(J26)/VPMOD(J26)-PHASQ(J26)/ROBS
      SLOW=T(IT)/ROBS
      DIDDLE=SLOW-1.0/VPMOD(J26)
      PH2=ROBS*(OMMOD(J26)*DIDDLE+PHASQ(J26)/ROBS)
      PHASQ(J26) = PHASQ(J26) - NHWR*(3.1415926535/2.0)
      CTRIG2=COS(PH2)
      STRIG2=SIN(PH2)
C
C THE INTEGRAL OF (AMPLTD * COS(OMEGA * (T - ROBS/VP) + PHAST)) OVER TH
C INTERVAL FROM OMMOD(KST(N)) TO OMMOD(KFIN(N)) IS FOUND BY SUMMING THE
C INTEGRALS FROM OMMOD(NOM-1) TO OMMOD(NOM) FOR NOM FROM KST(N)+1 TO
C KFIN(N)
      DO 50 NOM = NOST,NOFN
      A1 = A2
      PH1 = PH2
      CTRIG1=CTRIG2
      STRIG1=STRIG2
      S1=S2
      A2 = AMPLTD(NOM)*EXP(-AKIS(NOM)*ROBS)
      NHWR = RAD/3.1415926535
      PHASQ(NOM) = PHASQ(NOM) + NHWR*(3.1415926535/2.0)
      S2=OMMOD(NOM)/VPMOD(NOM)-PHASQ(NOM)/ROBS
      DIDDLE=SLOW-1.0/VPMOD(NOM)
      PH2=ROBS*(OMMOD(NOM)*DIDDLE+PHASQ(NOM)/ROBS)
      PHASQ(NOM) = PHASQ(NOM) - NHWR*(3.1415926535/2.0)
      OMEG1=OMMOD(NOM-1)
      OMEG2=OMMOD(NOM)
      DELPH = ROBS * ( SLOW * ( OMEG2 - OMEG1 ) - ( S2 - S1 ) )
      CALL AKI(OMEG1,OMEG2,A1,A2,CTRIG1,STRIG1,CTRIG2,STRIG2,
     1 DELPH,AKIINT)
```

```
   50 TNINT(N,IT) = TNINT(N,IT) + AKIINT
C
C PRESSURE IS EQUAL TO CF * ( VALUE OF OMEGA INTEGRAL )
   51 TNINT(N,IT) = CF * TNINT(N,IT)
C
C IF IOPT.LE.10 ALL THAT IS REQUESTED IS THE MODE IOPT RESPONSE, WHICH
C HAS JUST BEEN CALCULATED
      IF (IOPT.LE.10) GO TO 101
C
C INCREASE MODE NUMBER BY ONE
      N = N + 1
C
C IF N IS GREATER THAN MOFNO, ALL MODAL RESPONSES HAVE BEEN DETERMINED
      IF (N.LE.MOFNO) GO TO 11
C
C FOR EACH TIME IN T SET TOTAL PRESSURE EQUAL TO SUM OF MODAL PRESURES
      DO 60 IT=1,L
      DO 53 N = 1,MOFNO
   53 TOTINT(IT) = TOTINT(IT) + TNINT(N,IT)
      IF(IOPT.EQ. 11) GO TO 55
C
C WRITE TIME AND CORRESPONDING TOTAL ACOUSTIC RESPONSE (DYNES/CM**2)
      WRITE (6,54) T(IT),TOTINT(IT)
   54 FORMAT (1H ,49X,F9.1,10X,F12.2)
C
C WHEN IOPT.EQ.12 ONLY TOTAL RESPONSE IS PRINTED
      IF (IOPT.EQ.12) GO TO 59
C
C WHEN IOPT.EQ.11 ALL MODAL RESPONSES ARE ALSO PRINTED
   55 MM = MIN0(MOFNO,5)
      WRITE (6,57) IT,T(IT),TOTINT(IT),(TNINT(N,IT),N=1,MM)
   57 FORMAT (1H ,7X,I4,10X,F9.1,5X,F12.4,4X,F12.4,4X,F12.4,4X,F12.4,
     1  4X,F12.4,4X,F12.4)
   59 CONTINUE
   60 CONTINUE
      IF (MOFNO .LE. 5 .OR. IOPT .NE. 11 ) GO TO 65
      WRITE (6,61)
   61 FORMAT (1H0,20X,4HTIME,12X,5HTOTAL,11X,6HMODE 6,10X,6HMODE 7,10X,
     1  6HMODE 8,10X,6HMODE 9,10X,7HMODE 10/)
      DO 63 IT=1,L
   63 WRITE (6,57)  IT,T(IT),TOTINT(IT),(TNINT(N,IT),N=6,MOFNO)
C
   65 CONTINUE
   66 CALL PLOT(2.,.3,-3)
C     SIZE IS THE NUMBER OF SECONDS PER INCH IN THE PLOT
      SIZE = (T(L)-T(1))/600.
      IF (IOPT.LE.10) GO TO 107
      CALL SCALE(TOTINT,3.0,L,1)
C     AFTER SCALE RETURNS, TOTINT(L+1) IS THE MINIMUM VALUE OF THE
C     FIRST L VALUES.
C     TOTINT(L+2) IS (MAX-MIN)/3.0 OF THE FIRST L VALUES OF TOTINT
C     UMIN IS MAX-MIN OF TOTINT
      UMIN=TOTINT(L+2)*3.0
      UMIN = AINT(UMIN/25) * 25.0
      UMIN=AMAX1(UMIN,25.)
C     AT THIS POINT DY IS THE TOTAL RANGE IN TOTINT MOD25
      DY = ABS(UMIN)
      TOTINT(L+2)=DY/3.0
      DY=DY/3.0
C
C IF IOPT.EQ.12 PLOT ONLY THE TOTAL ACOUSTIC RESPONSE
      IF (IOPT.EQ.12) GO TO 70
C
C DRAW PRESSURE AXIS
```

```
      CALL PLOT(0.,0.,3)                                                TMPT
      ABC = MOFNO                                                       TMPT
      CALL PLOT(ABC,0.,2)                                               TMPT
      DO 69 N=1,MOFNO                                                   TMPT
      DO 67 J=1,L                                                       TMPT
   67 Y(J) = -1 * TNINT(N,J)                                            TMPT
   68 CALL PLOT(1.,0.,-3)                                               TMPT
C                                                                       TMPT
C PLOT ACOUSTIC RESPONSE (DYNES/CM**2) OF MODE N VERSUS TIME (SEC)      TMPT
      Y(L+1)=0.0                                                        TMPT
      Y(L+2)=TOTINT(L+2)                                                TMPT
      T(L+1)=T(1)                                                       TMPT
      T(L+2)=600.                                                       TMPT
   69 CALL LINE(Y,T,L,1,0,0)                                            TMPT
C                                                                       TMPT
   70 DO 73 J=1,L                                                       TMPT
   73 Y(J) = (-1) * TOTINT(J)                                           TMPT
C                                                                       TMPT
C DRAW PRESSURE AXIS                                                    TMPT
   75 CALL PLOT(0.,0.,3)                                                TMPT
      CALL PLOT(3.,0.,2)                                                TMPT
      CALL PLOT(1.5,0.,-3)                                              TMPT
      CALL NUMBER(.9,-.15,.15,DY,180.,0)                                TMPT
      CALL SYMBOL(.4,-.15,.15,"MICROBARS PER INCH",180.,18)             TMPT
      CALL AXIS(1.5,0.," ",1,SIZE,90.,T(1),600.)                        TMPT
      CALL SYMBOL(1.8,2.,.15,"TIME (SEC)",90.,10)                       TMPT
      Y(L+1)=0.0                                                        TMPT
      Y(L+2)=TOTINT(L+2)                                                TMPT
      T(L+1)=T(1)                                                       TMPT
      T(L+2)=600.                                                       TMPT
      CALL LINE(Y,T,L,1,0,0)                                            TMPT
      CALL PLOT(5.,-.3,-3)                                              TMPT
      GO TO 200                                                         TMPT
C                                                                       TMPT
C PRINT HISTORY OF MODE IOPT ONLY                                       TMPT
  101 WRITE (6,102) IOPT                                                TMPT
  102 FORMAT (1H1,45X,19HTABULATION OF MODE ,I2, 9H RESPONSE///1H ,48X, TMPT
     1  10HTIME (SEC),9X,15HP (DYNES/CM**2)//)                          TMPT
      DO 103 IT=1,L                                                     TMPT
  103 WRITE (6,104) T(IT),TNINT(IOPT,IT)                                TMPT
  104 FORMAT (1H ,49X,F9.1,10X,F12.4)                                   TMPT
      GO TO 66                                                          TMPT
C                                                                       TMPT
C     IF IOPT.LT.11 PLOT ONLY ACOUSTIC RESPONSE OF MODE IOPT            TMPT
  107 DO 108 J=1,L                                                      TMPT
  108 Y(J)=(-1)*TNINT(IOPT,J)                                           TMPT
C                                                                       TMPT
C     DETERMINE SCALE FOR PRESSURE AXIS WHEN IOPT.LT.11                 TMPT
  111 CALL SCALE(Y,2.0,L,1)                                             TMPT
      UMIN=Y(L+2)*2.0                                                   TMPT
      UMIN=AINT(UMIN/25)*25.0                                           TMPT
      UMIN=AMAX1(UMIN,25.)                                              TMPT
      DY=ABS(UMIN)                                                      TMPT
      Y(L+2)=DY/2.0                                                     TMPT
      DY=DY/2.0                                                         TMPT
      GO TO 75                                                          TMPT
C                                                                       TMPT
  200 RETURN                                                            TMPT
      END                                                               TMPT
```

```
      SUBROUTINE TCTINT(OMEGA,AKX,AKY,IT,L,XINT,PHI1,PHI2)               TOTINT
C     SUSPCT (SUBROUTINE)                    7/19/68   LAST CARD IN DECK IS TOTINT
C                                                                        TOTINT
C                                                                        TOTINT
C                        ----ABSTRACT----                                TOTINT
C                                                                        TOTINT
C TITLE - SUSPCT                                                         TOTINT
C       EVALUATION OF SUSPICION INDEX OF ELEMENT (N,M) OF MATRIX INMOD   TOTINT
C                                                                        TOTINT
C             SUSPCT EVALUATES THE SUSPICION INDEX, ISUS, OF THE ELEME   TOTINT
C             IN ROW N, COLUMN M OF THE MATRIX INMODE ( (N,M) MUST BE    TOTINT
C             AN INTERIOR ELEMENT).  THE NEIGHBORS OF (N,M) ARE DEFINE   TOTINT
C             TO BE THE EIGHT ELEMENTS WHICH FORM THE THREE BY THREE     TOTINT
C             ELEMENT SQUARE WHICH HAS (N,M) AT ITS CENTER.  THEY ARE    TOTINT
C             NUMBERED FROM ONE TO NINE BEGINNING IN THE UPPER LEFT AN   TOTINT
C             PROCEEDING CLOCKWISE (NO. 1 AND NO. 9 ARE SAME ELEMENT).   TOTINT
C             EACH ELEMENT OF MATRIX INMODE MUST HAVE ONE OF THREE       TOTINT
C             VALUES, -1, 1, OR 5.  (N,M) IS NOT SUSPICIOUS AND ISUS =   TOTINT
C             0 IF ANY ONE OF THE FOLLOWING CONDITIONS HOLDS.            TOTINT
C                                                                        TOTINT
C                 1.  ELEMENT (N,M) = 5                                  TOTINT
C                 2.  ANY OF ITS NEIGHBORS = 5                           TOTINT
C                 3.  NOWHERE IN THE 3X3 ARRAY OF (N,M) AND ITS NEIGH    TOTINT
C                     BORS DOES THERE APPEAR TO BE A DISPERSION CURVE    TOTINT
C                     WITH POSITIVE SLOPE                                TOTINT
C                                                                        TOTINT
C             OTHERWISE ISUS IS SET EQUAL TO THE NUMBER OF THE QUADRAN   TOTINT
C             IN WHICH THE POSITIVE SLOPE APPEARS.  THE QUADRANTS ARE    TOTINT
C             NUMBERED BEGINNING IN THE UPPER LEFT AND PROCEDING CLOCK   TOTINT
C             WISE.                                                      TOTINT
C                                                                        TOTINT
C LANGUAGE  - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)              TOTINT
C                                                                        TOTINT
C AUTHORS   - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                  TOTINT
C                                                                        TOTINT
C                        ----USAGE----                                   TOTINT
C                                                                        TOTINT
C     NO FORTRAN SUBROUTINES ARE CALLED                                  TOTINT
C                                                                        TOTINT
C FORTRAN USAGE                                                          TOTINT
C     CALL SUSPCT(N,M,NROW,INMODE,ISUS)                                  TOTINT
C                                                                        TOTINT
C INPUTS                                                                 TOTINT
C                                                                        TOTINT
C    N        ROW NUMBER OF ELEMENT UNDER CONSIDERATION (MAY NOT BE      TOTINT
C      I*4    FIRST OR LAST ROW)                                         TOTINT
C                                                                        TOTINT
C    M        COLUMN NUMBER OF ELEMENT UNDER CONSIDERAION (MAY NOT BE    TOTINT
C      I*4    FIRST OR LAST COLUMN)                                      TOTINT
C                                                                        TOTINT
C    NROW     TOTAL NUMBER OF ROWS IN INMODE                             TOTINT
C      I*4                                                               TOTINT
C                                                                        TOTINT
C    INMODE   MATRIX UNDER CONSIDERATION STORED IN VECTOR FORM, COLUMN   TOTINT
C     I*4(D)  AFTER COLUMN.  EACH ELEMENT MUST BE -1, 1, OR 5.           TOTINT
C                                                                        TOTINT
C OUTPUTS                                                                TOTINT
C                                                                        TOTINT
C    ISUS     SUSPICION INDEX OF ELEMENT (N,M).  SEE ABSTRACT ABOVE FO   TOTINT
C      I*4    DEFINITION.                                                TOTINT
C                                                                        TOTINT
C                                                                        TOTINT
C                        ----EXAMPLES----                                TOTINT
C                                                                        TOTINT
```

```
C CALLING PROGRAM
C
C     DIMENSION INMODE(9)
C     INMODE = -1, -1, 1, 1, -1, 1, 1, 1, -1
C     CALL SUSFCT(2,2,3,INMODE,ISUS)
C     WRITE (6,200) ISUS
C 200 FORMAT (10H EXAMPLE 1,6X, 6HISUS =,I2)
C     INMODE = -1, -1, 1, 1, -1, -1, 1, 1, 1
C     CALL SUSFCT(2,2,3,INMODE,ISUS)
C     WRITE (6,300) ISUS
C 300 FORMAT (10H EXAMPLE 2,6X, 6HISUS =,I2)
C     END
C
C     TMPT (SUBROUTINE)                   7/19/68
C
C
C              ----ABSTRACT----
C
C TITLE - TMPT
C       CALCULATION AND PLOTTING OF FAR-FIELD TRANSIENT RESPONSE TO A
C       PRESSURE SOURCE IN THE ATMOSPHERE
C
C             THE RESPONSE OF MODE N IS FOUND BY INTEGRATING (AMPLTO *
C             COS( OMEGA * (T - R/VP) + PHASQ) OVER OMEGA FROM OMMOD
C             (KST(N)) TO OMMOD(KFIN(N)) AND DIVIDING BY SQRT(R). VP,
C             PHASQ, AND AMPLTO ARE FUNCTIONS OF BOTH N AND OMEGA.  TH
C             TOTAL RESPONSE IS THE SUM OF THE MODAL RESPONSES.  THE
C             RESPONSE IS CALCULATED FOR TIME TFIRST AND AT INTERVALS
C             OF DELTT THEREAFTER UNTIL TEND IS REACHED.  THE VALUE OF
C             IOPT DETERMINES WHAT WILL BE CALCULATED, PRINTED AND
C             PLOTTED.  (SEE INPUT LIST FOR POSSIBLE IOPT VALUES.) THE
C             RESULTS ARE TABULATED IN THE PRINTOUT AND GRAPHED BY THE
C             CALCOMP PLOTTER.
C
C             THE CURRENT VERSION OF THIS SUBROUTINE DIFFERS FROM THAT
C             REPORTED IN AFCRL-70-0134 IN THAT THE RESULTING THEORETI
C             CAL PRESSURE PERTURBATIONS INCLUDE THE EARTH CURVATURE
C             CORRECTION FACTOR, ( ROBS / ( RE * SIN(ROBS/RE) ) )**0.5
C
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C AUTHOR      - J.W.POSEY, M.I.T., JUNE,1968
C
C
C              ----USAGE----
C
C     FORTRAN SUBROUTINE AKI IS CALLED
C
C     CALCOMP PLOTTER SUBROUTINES PLOT1, AXIS1, NUMBR1, SYMBL5, AND
C     SCLGPH ARE CALLED TO WRITE THE CALCOMP TAPE.  SUBROUTINE NEWPLT
C     MUST HAVE BEEN CALLED PRIOR TO CALLING TMPT, AND ENDPLT MUST BE
C     CALLED AFTER RETURNING  FROM TMPT.  (SEE MAIN PROGRAM)
C
C FORTRAN USAGE
C     CALL TMPT(TFIRST,TEND,DELTT,ROBS,MODFND,KST,KFIN,OMMOD,VPMOD,AMPLT
C    1  ,PHASQ,IOPT)
C
C INPUTS
C
C     TFIRST    TIME AT WHICH TABULATION AND PLOTTING OF RESPONSE IS TO
C       R*4     BEGIN  (SEC)
C
C     TEND      TIME AT WHICH TABULATION AND PLOTTING OF RESPONSE IS TO
C      TOTINT (SUBROUTINE)               7/27/68   LAST CARD IN DECK IS
C
```

```
C                    ----ABSTRACT----
C
C TITLE - TOTINT
C        THIS SUBROUTINE COMPUTES THE TOTAL INTEGRAL
C
C          XINT = INTEGRAL OVER Z FROM 0 TO INFINITY OF
C
C                A3(Z)*(A1(Z)*F1(Z) + A2(Z)*F2(Z))**2                (1
C
C        THE ATMOSPHERE IS ASSUMED TO BE REPRESENTED IN A MULTILAYER FO
C        WITH A1,A2, AND A3 CONSTANT IN EACH LAYER.  THE INTEGRAL IS
C        EVALUATED AS A SUM OF INTEGRALS OVER INDIVIDUAL LAYERS.
C
C        THE FUNCTIONS F1(Z) AND F2(Z) ARE CONTINUOUS ACROSS LAYER
C        BOUNDARIES AND SATISFY THE RESIDUAL EQUATIONS
C
C           DF1(Z)/DZ = A(1,1)*F1(Z) + A(1,2)*F2(Z)                  (2
C           DF2(Z)/DZ = A(2,1)*F1(Z) + A(2,2)*F2(Z)                  (2
C
C        WHERE THE ELEMENTS OF THE MATRIX A (COMPUTED BY SUBROUTINE AAA
C        ARE CONSTANT IN EACH LAYER.
C
C        THE FUNCTIONS F1(Z) AND F2(Z) ARE ASSUMED TO SATISFY THE UPPER
C        BOUNDARY CONDITION THAT BOTH DECREASE EXPONENTIALLY WITH
C        INCREASING HEIGHT IN THE UPPER HALFSPACE.  THE NORMALIZATION
C        OF THE FUNCTIONS IS SUCH THAT AT THE LOWER BOUNDARY Z0 OF THE
C        UPPER HALFSPACE
C
C           F1(Z0)= -SQRT(G)*A(1,2)                                  (3
C           F2(Z0)= SQRT(G)*(G+A(1,1))                               (3
C
C        WITH
C
C           G = SQRT(A(1,1)**2 + A(1,2)*A(2,1))                      (4
C
C        THE ELEMENTS A(I,J) IN EQNS. (3) AND (4) ARE THOSE APPROPRIATE
C        TO THE UPPER HALFSPACE.  IF G**2 IS NEGATIVE, THE PROGRAM
C        RETURNS L=-1.  OTHERWISE IT RETURNS L=1.
C
C PROGRAM NOTES
C
C              THE INTEGRATION OVER THE UPPER HALFSPACE IS PERFORMED BY
C              CALLING UPINT.  THE INTEGRATIONS OVER THE LAYERS OF FINI
C              THICKNESS ARE PERFORMED BY CALLING ELINT.
C
C              THE PARAMETERS A1,A2,A3 DEPEND IN GENERAL ON ANGULAR
C              FREQUENCY OMEGA, HORIZONTAL WAVENUMBER COMPONENTS AKX
C              AND AKY, SOUND SPEED C, AND WIND VELOCITY COMPONENTS VX
C              AND VY.  THE FORMULAS USED ARE CONTROLLED BY THE INPUT
C              PARAMETER IT WHICH IS TRANSMITTED TO SUBROUTINE USEAS.
C
C              THE PARAMETERS DEFINING THE MULTILAYER ATMOSPHERE ARE
C              PRESUMED STORED IN COMMON
C
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)
C
C AUTHOR     - A.D.PIERCE, M.I.T., JULY,1968
C
C                ----CALLING SEQUENCE----
C
C SEE SUBROUTINE NAMPDE
C     DIMENSION CI(100),VXI(100),VYI(100),HI(100),PHI1(100),PHI2(100)
C     COMMON IMAX,CI,VXI,VYI,HI        (THESE MUST BE IN COMMON)
C     CALL TOTINT(OMEGA,AKX,AKY,IT,L,XINT,PHI1,PHI2)
```

```
C
C                ----EXTERNAL SUBROUTINES REQUIRED----
C
C    AAAA,MMMM,CAI,SAI,USEAS,UPINT,ELINT,BBBB
C
C    AAAA AND BBBB ARE CALLED BY ELINT.
C    CAI AND SAI ARE CALLED BY BBBB.
C
C                ----ARGUMENT LIST----
C
C    OMEGA       R*4    NO    INP
C    AKX         R*4    NO    INP
C    AKY         R*4    NO    INP
C    IT          I*4    NO    INP
C    L           I*4    NO    OUT
C    XINT        I*4    NO    OUT
C    PHI1        R*4    100   INP
C    PHI2        R*4    100   INP
C
C COMMON STORAGE USED
C    COMMON IMAX,CI,VXI,VYI,HI
C
C    IMAX        I*4    NO    INP
C    CI          R*4    100   INP
C    VXI         R*4    100   INP
C    VYI         R*4    100   INP
C    HI          R*4    100   INP
C
C                ----INPUTS----
C
C    OMEGA       =ANGULAR FREQUENCY IN RADIANS/SEC
C    AKX         =X COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)
C    AKY         =Y COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)
C    IT          =PARAMETER TRANSMITTED TO USEAS DEFINING FUNCTIONAL
C                 DEPENDENCE OF A1,A2,A3 COMPUTED BY USEAS.
C    PHI1(I)     =VALUE OF F1 AT BOTTOM OF LAYER I
C    PHI2(I)     =VALUE OF F2 AT BOTTOM OF LAYER I
C    IMAX        =NUMBER OF ATMOSPHERIC LAYERS WITH FINITE THICKNESS
C    CI(I)       =SOUND SPEED (KM/SEC) IN I-TH LAYER
C    VXI(I)      =X COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C    VYI(I)      =Y COMPONENT OF WIND VELOCITY (KM/SEC) IN I-TH LAYER
C    HI(I)       =THICKNESS IN KM OF I-TH LAYER
C
C                ----OUTPUTS----
C
C    L           =1 OR -1 DEPENDING ON WHETHER UPPER BOUNDARY CONDITIO
C                 CAN OR CANNOT BE SATISFIED.  SEE SUBROUTINE UPINT
C    XINT        =INTEGRAL OVER Z FROM 0 TO INFINITY AS DEFINED IN THE
C                 ABSTRACT.
C
C
C                ----PROGRAM FOLLOWS BELOW----
C
C
C
C DIMENSION AND COMMON STATEMENTS
      DIMENSION CI(100),VXI(100),VYI(100),HI(100),EM(2,2)
      DIMENSION PHI1(100),PHI2(100)
      COMMON IMAX,CI,VXI,VYI,HI
C
C COMPUTATION OF CONTRIBUTION FROM UPPER HALFSPACE
      J=IMAX+1
      C=CI(J)
      VX=VXI(J)
```

```
      VY=VYI(J)
      CALL USEAS(OMEGA,AKX,AKY,C,VX,VY,IT,A1,A2,A3)
      CALL UPINT(OMEGA,AKX,AKY,C,VX,VY,A1,A2,L,F1,F2,UINT)
C
C CHECK IF L NEGATIVE
      IF(L .LT. 0) RETURN
C
C WE DENOTE THE CONTRIBUTION A3*UINT BY XINT.  AS THE COMPUTATION CON-
C TINUES, XINT WILL SUCCESSIVELY REPRESENT THE VARIOUS SUBTOTALS UNTIL
C CONTRIBUTIONS FROM ALL THE LAYERS HAVE BEEN ADDED IN.
      XINT=A3*UINT
C
C START OF DO LOOP
      DO 90 I=1,IMAX
      J=IMAX+1-I
C
C COMPUTATION OF CONTRIBUTION FROM J-TH LAYER OF FINITE THICKNESS.
C THE CURRENT VALUES F1 AND F2 REPRESENT F1(Z) AND F2(Z) AT TOP OF
C J-TH LAYER.
      C=CI(J)
      VX=VXI(J)
      VY=VYI(J)
      H=HI(J)
      CALL USEAS(OMEGA,AKX,AKY,C,VX,VY,IT,A1,A2,A3)
      CALL ELINT(OMEGA,AKX,AKY,C,VX,VY,H,F1,F2,A1,A2,AINT)
      XINT=XINT+AINT*A3
C
C COMPUTATION OF F1 AND F2 APPROPRIATE TO TOP OF (J-1)-TH LAYER
      F1 = PHI1(J)
   90 F2 = PHI2(J)
C END OF DO LOOP
C
      RETURN
      END
```

```
      SUBROUTINE UPINT(OMEGA,AKX,AKY,C,VX,VY,A1,A2,L,F1,F2,UINT)
C     UPINT (SUBROUTINE)                  7/25/68   LAST CARD IN DECK IS
C
C                        ----ABSTRACT----
C
C TITLE - UPINT
C            THIS SUBROUTINE COMPUTES AN INTEGRAL OF THE FORM
C
C               UINT = INTEGRAL OVER Z FROM Z0 TO INFINITY OF
C
C                      (A1*F1(Z) + A2*F2(Z))**2                     (1
C
C            THE FUNCTIONS F1(Z) AND F2(Z) ARE THE SOLUTIONS OF THE COUPLED
C            ORDINARY DIFFERENTIAL EQUATIONS
C
C               DF1/DZ = A11*F1 + A12*F2                            (2
C               DF2/DZ = A21*F1 + A22*F2                            (2
C
C            WHERE THE ELEMENTS OF THE MATRIX A ARE INDEPENDENT OF Z.  THE
C            FUNCTIONS F1(Z) AND F2(Z) ARE SUBJECT TO THE UPPER BOUNDARY
C            CONDITION THAT BOTH DECREASE EXPONENTIALLY WITH INCREASING
C            ALTITUDE.  SINCE THE MATRIX A IS COMPUTED BY AAAA, INSURING
C            A(2,2)=-A(1,1), BOTH SHOULD VARY WITH HEIGHT AS EXP(-G*(Z-Z0))
C            WHERE
C
C               G = SQRT(A(1,1)**2+A(1,2)*A(2,1))                   (3
C
C            IT IS ASSUMED G**2 IS POSITIVE.  OTHERWISE L=-1 IS RETURNED.
C
C            THE EXPLICIT FORMS ADOPTED FOR F1 AND F2 WHICH SATISFY (2) ARE
C
C               F1 =-SQRT(G)*A(1,2)*EXP(-G*(Z-Z0))                  (4
C               F2 = SQRT(G)*(G+A(1,1))*EXP(-G*(Z-Z0))              (4
C
C            THUS UINT IS GIVEN BY
C
C               UINT =((-A1*A(1,2)+A2*(G+A(1,1)))**2)/2.0           (5
C
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL C22-6515-4)
C AUTHOR      - A.D.PIERCE, M.I.T., JULY,1968
C
C                        ----CALLING SEQUENCE----
C
C SEE SUBROUTINE TOTINT
C NO DIMENSION STATEMENTS REQUIRED
C     CALL UPINT(OMEGA,AKX,AKY,C,VX,VY,A1,A2,L,F1,F2,UINT)
C
C                        ----EXTERNAL SUBROUTINES REQUIRED----
C
C     AAAA
C
C                        ----ARGUMENT LIST----
C
C     OMEGA       R*4     NO     INP
C     AKX         R*4     NO     INP
C     AKY         R*4     NO     INP
C     C           R*4     NO     INP
C     VX          R*4     NO     INP
C     VY          R*4     NO     INP
C     A1          R*4     NO     INP
C     A2          R*4     NO     INP
C     L           I*4     NO     OUT
C     F1          R*4     NO     OUT
C     F2          R*4     NO     OUT
```

```
C     UINT        R*4     NO     OUT
C
C NO COMMON STORAGE USED
C
C                    ----INPUTS----
C
C     OMEGA        =ANGULAR FREQUENCY IN RADIANS/SEC
C     AKX          =X COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)
C     AKY          =Y COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)
C     C            =SOUND SPEED IN KM/SEC
C     VX           =X COMPONENT OF WIND VELOCITY IN KM/SEC
C     VY           =Y COMPONENT OF WIND VELOCITY IN KM/SEC
C     A1           =COEFFICIENT OF F1(Z) IN INTEGRAND
C     A2           =COEFFICIENT OF F2(Z) IN INTEGRAND
C
C                    ----OUTPUTS----
C
C     L            =1 OR -1 DEPENDING ON WHETHER UPPER BOUNDARY CONDITIO
C                   OF F1(Z),F2(Z) DECREASING EXPONENTIALLY WITH INCREAS
C                   HEIGHT CAN OR CANNOT BE SATISFIED.  IT REPRESENTS TH
C                   SIGN OF G**2 WHERE G IS DEFINED IN THE ABSTRACT.
C     F1           =VALUE OF F1(Z) AT BOTTOM OF HALFSPACE, DEFINED AS
C                   -SQRT(G)*A(1,2) FROM EQN. (4A).
C     F2           =VALUE OF F2(Z) AT BOTTOM OF HALFSPACE, DEFINED AS
C                   SQRT(G)*(G+A(1,1)) FROM EQN. (4B)
C     UINT         =THE INTEGRAL DEFINED BY EQNS. (1) AND (5) IN THE
C                   ABSTRACT
C
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
      DIMENSION A(2,2)
C
C COMPUTATION OF A MATRIX AND OF X=G**2
      CALL AAAA(OMEGA,AKX,AKY,C,VX,VY,A)
      X=A(1,1)**2+A(1,2)*A(2,1)
C CHECK ON SIGN OF X
    2 IF( X .GT. 0.0 ) GO TO 3
C
C X IS NEGATIVE
      L=-1
      RETURN
C CONTINUING FROM 2 WITH X POSITIVE
    3 L=1
      G=SQRT(X)
      GRT=SQRT(G)
      F1=-GRT*A(1,2)
      F2=GRT*(G+A(1,1))
C COMPUTATION OF UINT
      UINT=(-A1*A(1,2)+A2*(G+A(1,1)))**2/2.0
      RETURN
      END
```

```
      SUBROUTINE USEAS(OMEGA,AKX,AKY,C,VX,VY,IT,A1,A2,A3)               USEAS
C     USEAS (SUBROUTINE)                   7/25/68   LAST CARD IN DECK IS USEAS
C                                                                        USEAS
C                   ----ABSTRACT----                                     USEAS
C                                                                        USEAS
C TITLE - USEAS                                                          USEAS
C          THE PURPOSE OF THIS SUBROUTINE IS TO COMPUTE THE NUMBERS A1, A USEAS
C          AND A3 WHICH DEPEND ON ANGULAR FREQUENCY OMEGA, HORIZONTAL WAV USEAS
C          NUMBER COMPONENTS AKX AND AKY, THE SOUND SPEED C, AND THE WIND USEAS
C          SPEED COMPONENTS VX AND VY.  THE INTEGER IT DETERMINES WHICH  USEAS
C          FORMULAS ARE USED FOR A1, A2, AND A3 ACCORDING TO THE FOLLOWIN USEAS
C          TABLE                                                         USEAS
C                                                                        USEAS
C             (IT)        (A1)      (A2)               (A3)              USEAS
C          ------------------------------------------------------------- USEAS
C              1           1         0                  1                USEAS
C              2           0         1                  1                USEAS
C              3           1         0            BOM*(KDOTV)/(C**2*K)   USEAS
C              4           1         0            BOM/C**2               USEAS
C              5           1         0            VX*BOM/C**2            USEAS
C              6           1         0            VY*BOM/C**2            USEAS
C              7          G/C       -C            K*OMEGA/BOM**3         USEAS
C              8          G/C       -C            1.0/BOM**2             USEAS
C              9          G/C       -C            K**2/BOM**3            USEAS
C             10          G/C       -C            VX*K**2/BOM**3         USEAS
C             11          G/C       -C            VY*K**2/BOM**3         USEAS
C                                                                        USEAS
C          HERE BOM=OMEGA-KDOTV IS THE DOPPLER SHIFTED ANGULAR FREQUENCY. USEAS
C          KDOTV=AKX*VX+AKY*VY IS THE DOT PRODUCT OF WAVE NUMBER WITH    USEAS
C          THE WIND VELOCITY, AND K=SQRT(AKX**2+AKY**2) IS THE MAGNITUDE USEAS
C          OF THE WAVE NUMBER VECTOR.  THE ACCELERATION OF GRAVITY G IS  USEAS
C          TAKEN AS .0098 KM/SEC**2 IN THE COMPUTATION.  COMPUTED VALUES USEAS
C          SHOULD BE IN KM,SEC SYSTEM OF UNITS.                          USEAS
C                                                                        USEAS
C LANGUAGE   - FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)             USEAS
C AUTHOR     - A.D.PIERCE, M.I.T., JUNE,1968                             USEAS
C                                                                        USEAS
C                   ----CALLING SEQUENCE----                             USEAS
C                                                                        USEAS
C SEE SUBROUTINE TOTINT                                                  USEAS
C     NO DIMENSION STATEMENTS ARE REQUIRED                               USEAS
C     IT=                                                                USEAS
C     CALL USEAS(OMEGA,AKX,AKY,C,VX,VY,IT,A1,A2,A3)                      USEAS
C A1,A2,A3 ARE NOW AVAILABLE FOR FUTURE COMPUTATIONS                     USEAS
C                                                                        USEAS
C NO EXTERNAL LIBRARY SUBROUTINES ARE REQUIRED                           USEAS
C                                                                        USEAS
C                ----ARGUMENT LIST----                                   USEAS
C                                                                        USEAS
C     OMEGA      R*4     NO     INP                                      USEAS
C     AKX        R*4     NO     INP                                      USEAS
C     AKY        R*4     NO     INP                                      USEAS
C     C          R*4     NO     INP                                      USEAS
C     VX         R*4     NO     INP                                      USEAS
C     VY         R*4     NO     INP                                      USEAS
C     IT         I*4     NO     INP                                      USEAS
C     A1         R*4     NO     OUT                                      USEAS
C     A2         R*4     NO     OUT                                      USEAS
C     A3         R*4     NO     OUT                                      USEAS
C                                                                        USEAS
C NO COMMON STORAGE USED                                                 USEAS
C                                                                        USEAS
C                   ----INPUTS----                                       USEAS
C                                                                        USEAS
```

```
C     OMEGA        =ANGULAR FREQUENCY IN RAD/SEC                         USEAS
C     AKX          =X COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)        USEAS
C     AKY          =Y COMPONENT OF WAVE NUMBER VECTOR IN KM**(-1)        USEAS
C     C            =SOUND SPEED IN KM/SEC                                USEAS
C     VX           =X COMPONENT OF WIND VELOCITY IN KM/SEC               USEAS
C     VY           =Y COMPONENT OF WIND VELOCITY IN KM/SEC               USEAS
C     IT           =CONTROL PARAMETER FOR SELECTION OF FORMULAS (SEE     USEAS
C                   ABSTRACT).                                           USEAS
C                                                                        USEAS
C                  ----OUTPUTS----                                       USEAS
C                                                                        USEAS
C     A1           =PARAMETER DEFINED BY FORMULAS IN ABSTRACT            USEAS
C     A2           =PARAMETER DEFINED BY FORMULAS IN ABSTRACT            USEAS
C     A3           =PARAMETER DEFINED BY FORMULAS IN ABSTRACT            USEAS
C                                                                        USEAS
C                                                                        USEAS
C                  ----PROGRAM FOLLOWS BELOW----                         USEAS
C                                                                        USEAS
C                                                                        USEAS
C                                                                        USEAS
C WE ASSIGN VALUES TO A1,A2,A3 WHICH WILL NOT NECESSARILY BE THEIR EXIT USEAS
C VALUES.                                                                USEAS
      A1=1.0                                                             USEAS
      A2=0.0                                                             USEAS
      A3=1.0                                                             USEAS
C IF IT IS 1, THESE ARE CORRECT. HOWEVER.                                USEAS
      IF(IT .EQ. 1) RETURN                                               USEAS
      IF(IT .GT. 2) GO TO 200                                            USEAS
C                                                                        USEAS
C IT IS 2.  THE CURRENT VALUES ARE 1,0,1.  WE CHANGE THE FIRST TWO.      USEAS
      A1=0.0                                                             USEAS
      A2=1.0                                                             USEAS
      RETURN                                                             USEAS
C                                                                        USEAS
C IT IS .GT. 2.  WE COMPUTE SOME QUANTITIES FOR FUTURE REFERENCE         USEAS
  200 AKV=AKX*VX+AKY*VY                                                  USEAS
      AKSQ=AKX**2+AKY**2                                                 USEAS
      BOM=OMEGA-AKV                                                      USEAS
      AK=SQRT(AKSQ)                                                      USEAS
C THE CURRENT VALUES OF A1,A2,A3 ARE STILL 1,0,1.                        USEAS
      IF(IT .GT. 3) GO TO 300                                            USEAS
C                                                                        USEAS
C IT IS EQUAL TO 3.  ONLY A3 NEED BE CHANGED.                            USEAS
      A3=BOM*AKV/(C**2*AK)                                               USEAS
      RETURN                                                             USEAS
C                                                                        USEAS
C IT IS 4 OR GREATER.  WE SET A3 TO VALUE APPROPRIATE FOR IT=4.          USEAS
  300 A3=BOM/C**2                                                        USEAS
C THE CURRENT VALUES OF A1 AND A2 ARE 1 AND 0.                           USEAS
      IF(IT .EQ. 4) RETURN                                               USEAS
      IF(IT .EQ. 5) A3=VX*A3                                             USEAS
      IF(IT .EQ. 6) A3=VY*A3                                             USEAS
      IF(IT .EQ. 5 .OR. IT .EQ. 6) RETURN                                USEAS
C                                                                        USEAS
C IT IS 7 OR LARGER                                                      USEAS
      A1=.0098/C                                                         USEAS
      A2=-C                                                              USEAS
C                                                                        USEAS
C THE ONLY QUANTITY WE NEED DETERMINE IS A3                              USEAS
C                                                                        USEAS
      IF(IT .GT. 7) GO TO 700                                            USEAS
C IT=7                                                                   USEAS
      A3=AK*OMEGA/BOM**3                                                 USEAS
      RETURN                                                             USEAS
```

```
C
  700 IF(IT .GT.9) GO TO 800
C IT=8
      A3=1.0/BOM**2
      RETURN
C
C FOR IT=9,10,11 WE NEED THE FACTOR AKSQ/BOM**3
  800 A3=AKSQ/BOM**3
      IF(IT .EQ. 9) RETURN
      IF(IT .GT. 10) GO TO 1000
C IT=10
      A3=VX*A3
      RETURN
C
C IT=11 (YOU SHOULDN-T INPUT IT FOR VALUES OUTSIDE RANGE OF 1 TO 11.)
 1000 A3=VY*A3
      RETURN
      END
```

```
      SUBROUTINE WIDEN(OM,V,INMODE,NOM,NOMP,NVP,N1,KW,THETK)             WIDEN
C     WIDEN (SUBROUTINE)                    6/19/68   LAST CARD IN DECK IS WIDEN
C                                                                        WIDEN
C                                                                        WIDEN
C                        ----ABSTRACT----                                WIDEN
C                                                                        WIDEN
C TITLE - WIDEN                                                          WIDEN
C        WIDEN MATRIX INMODE BY ADDING KW COLUMNS BETWEEN COLUMNS N1 AN WIDEN
C        N1+1                                                            WIDEN
C                                                                        WIDEN
C               WIDEN ADDS KW ELEMENTS TO THE VECTOR OF ANGULAR FREQUENC WIDEN
C               OM , DIVIDING THE INTERVAL BETWEEN OM(N1) AND OM(N1+1) I WIDEN
C               KW+1 EQUAL PARTS.  FOR EACH NEW ANGULAR FREQUENCY, A NEW WIDEN
C               COLUMN IS ADDED TO THE INMODE MATRIX (DEFINED IN SUBROU- WIDEN
C               TINE MPOUT).  INMODE IS STORED IN VECTOR FORM, COLUMN AF WIDEN
C               COLUMN.                                                  WIDEN
C                                                                        WIDEN
C LANGUAGE   -- FORTRAN IV (360, REFERENCE MANUAL C28-6515-4)            WIDEN
C                                                                        WIDEN
C AUTHORS    - A.D.PIERCE AND J.POSEY, M.I.T., JUNE,1968                 WIDEN
C                                                                        WIDEN
C                                                                        WIDEN
C                        ----USAGE----                                   WIDEN
C                                                                        WIDEN
C     OM,V,INMODE MUST BE DIMENSIONED IN CALLING PROGRAM                 WIDEN
C     THE ONLY SUBROUTINE CALLED IS NMOFN, DESCRIBED ELSEWHERE IN THIS   WIDEN
C     SERIES                                                             WIDEN
C                                                                        WIDEN
C FORTRAN USAGE                                                          WIDEN
C     CALL WIDEN(OM,V,INMODE,NOM,NOMP,NVP,N1,KW,THETK)                   WIDEN
C                                                                        WIDEN
C INPUTS                                                                 WIDEN
C                                                                        WIDEN
C    OM         VECTOR WHOSE ELEMENTS ARE THE VALUES OF ANGULAR FREQUENC WIDEN
C      R*4(D)   CORRESPONDING TO THE COLUMNS OF MATRIX INMODE. (RAD/SEC) WIDEN
C                                                                        WIDEN
C    V          VECTOR WHOSE ELEMENTS ARE THE VALUES OF PHASE VELOCITY   WIDEN
C      R*4(D)   CORRESPONDING TO THE ROWS OF MATRIX INMODE. (KM/SEC)     WIDEN
C                                                                        WIDEN
C    INMODE     EACH ELEMENT OF THIS MATRIX CORRESPONDS TO A POINT IN TH WIDEN
C      I*4(D)   FREQUENCY (OM) - PHASE VELOCITY (V) PLANE.  IF THE NORMA WIDEN
C               MODE DISPERSION FUNCTION (FPP, FOUND BY CALLING SUBROUTI WIDEN
C               NMOFN) IS POSITIVE AT THAT POINT, THE ELEMENT IS +1. IF  WIDEN
C               FPP IS NEGATIVE, THE ELEMENT IS -1. IF FPP DOES NOT EXIS WIDEN
C               THE ELEMENT IS 5.  INMODE IS STORED IN VECTOR FORM, COLU WIDEN
C               AFTER COLUMN.                                            WIDEN
C                                                                        WIDEN
C    NOM        NUMBER OF ELEMENTS IN OM (AND NO. OF COLUMNS IN INMODE)  WIDEN
C      I*4      WHEN WIDEN IS CALLED.                                    WIDEN
C    NVP        NUMBER OF ELEMENTS IN V (AND NO. OF ROWS IN INMODE).     WIDEN
C      I*4                                                               WIDEN
C                                                                        WIDEN
C    N1         NUMBER OF INMODE COLUMN IMMEDIATELY LEFT OF SPACE IN WHI WIDEN
C      I*4      NEW COLUMNS ARE TO BE ADDED.                             WIDEN
C                                                                        WIDEN
C    KW                                                                  WIDEN
C      I*4      NUMBER OF COLUMNS TO BE ADDED TO INMODE.                 WIDEN
C                                                                        WIDEN
C    THETK      PHASE VELOCITY DIRECTION MEASURED COUNTER-CLOCKWISE FROM WIDEN
C      R*4      X-AXIS (RADIANS).                                        WIDEN
C                                                                        WIDEN
C OUTPUTS                                                                WIDEN
C                                                                        WIDEN
C    THE OUTPUTS ARE NOMP (= NOM + KW = THE NEW NUMBER OF ELEMENTS IN O WIDEN
```

```
C    AND THE NEW NUMBER OF COLUMNS IN INMODE) AND REVISED VERSIONS OF O WIDEN
C    AND INMODE.                                                        WIDEN
C                                                                       WIDEN
C                                                                       WIDEN
C                    ----EXAMPLE----                                    WIDEN
C                                                                       WIDEN
C    SUPPOSE OM = 1.0,2.0,3.0 AND WIDEN IS CALLED WITH KW = 3, AND N1 = WIDEN
C    2, THEN UPON RETURN TO CALLING PROGRAM, OM = 1.0,2.0,2.25,2.5,2.75 WIDEN
C    3.0, NOMP = 6, AND INMODE WILL HAVE THREE NEW ROWS CORRESPONDING T WIDEN
C    THE NEW ELEMENTS OF OM.                                            WIDEN
C                                                                       WIDEN
C                                                                       WIDEN
C                  ----PROGRAM FOLLOWS BELOW----                        WIDEN
C                                                                       WIDEN
C                                                                       WIDEN
C                                                                       WIDEN
C VARIABLE DIMENSIONING                                                 WIDEN
      DIMENSION OM(1),V(1),INMODE(1)                                    WIDEN
      COMMON IMAX,CI(100),VXI(100),VYI(100),HI(100)                     WIDEN
C                                                                       WIDEN
C INTERVAL AT WHICH NEW VALUES OF OM ARE BE PLACED BETWEEN OM(N1) AND   WIDEN
C OM(N1+1) IS DETERMINED                                                WIDEN
      DELOM=(OM(N1+1)-OM(N1))/(KW+1)                                    WIDEN
C                                                                       WIDEN
C NOMP IS NUMBER OF ELEMENT IN EXPANDED OM                              WIDEN
      NOMP=NOM+KW                                                       WIDEN
C                                                                       WIDEN
C NSTART IS THE NUMBER OF THE ELEMENT IN THE NEW OM WHICH CORRESPONDS T WIDEN
C ELEMENT N1+1 IN THE OLD OM VECTOR                                     WIDEN
      NSTART=N1+1+KW                                                    WIDEN
C                                                                       WIDEN
C MOVE ALL ELEMENTS OF OM BEYOND ELEMENT N1 TO THEIR NEW POSITIONS, BEG WIDEN
C NING WITH THE LAST ELEMENT                                            WIDEN
      DO 90 NJ=NSTART,NOMP                                              WIDEN
      J=NOMP-(NJ-NSTART)                                                WIDEN
      JOLD=J-KW                                                         WIDEN
C                                                                       WIDEN
C MOVE COLUMN JOLD OF INMODE INTO POSITION FOR COLUMN J                 WIDEN
      OM(J)=OM(JOLD)                                                    WIDEN
      DO 90 IP=1,NVP                                                    WIDEN
      IJ=(J-1)*NVP+(NVP-IP) + 1                                         WIDEN
      IJOLD=(JOLD-1)*NVP+(NVP-IP) + 1                                   WIDEN
      INMODE(IJ)=INMODE(IJOLD)                                          WIDEN
   90 CONTINUE                                                          WIDEN
C                                                                       WIDEN
C NSTART IS NUMBER OF FIRST NEW COLUMN                                  WIDEN
      NSTART=N1+1                                                       WIDEN
C                                                                       WIDEN
C NEND IS NUMBER OF LAST NEW COLUMN                                     WIDEN
      NEND=N1+KW                                                        WIDEN
C                                                                       WIDEN
C NEW VALUES OF OM ARE ESTABLISHED                                      WIDEN
      OMEGA=OM(N1)                                                      WIDEN
      DO 190 J=NSTART,NEND                                              WIDEN
      OM(J)=OMEGA + DELOM                                               WIDEN
      OMEGA = OM(J)                                                     WIDEN
      DO 190 I=1,NVP                                                    WIDEN
C                                                                       WIDEN
C IJ IS NUMBER OF ELEMENT IN VECTOR REPRESENTATION OF INMODE WHICH IS   WIDEN
C ELEMENT J IN ROW I OF INMODE                                          WIDEN
      IJ=(J-1)*NVP+I                                                    WIDEN
      VPHSE=V(I)                                                        WIDEN
C                                                                       WIDEN
C CALL NMDFN TO EVALUATE THE NORMAL MODE DISPERSION FUNCTION (FPP)      WIDEN
```

```
      CALL NMOFN(OMEGA,VPHSE,THETK,L,FFP,K)
C
C IF FPP DOES NOT EXIST L = -1
      IF( L .EQ. -1 ) GO TO 150
C
C IF FPP DOES EXIST L = 1 AND INMODE(IJ) = (FPP/ABS(FPP))
      INMODE(IJ) = 1
      IF (FPP.LE.0.0) INMODE(IJ) = -1
      GO TO 180
  150 INMODE(IJ)=5
  180 CONTINUE
  190 CONTINUE
      RETURN
      END
```

APPENDIX B

SOURCE DECK LISTING OF AN ALTERNATE VERSION OF SUBROUTINE TABLE

This version of SUBROUTINE TABLE is used, as described in Chapter III of the present report, to tabulate listings of R_{11} and R_{12} versus angular frequency OMEGA and phase velocity VPHSE which are used in calculating the parameter α and β for the GR_0 and GR_1 modes which in turn are used in calculating the values of the imaginary component k_I of horizontal wave-number for these modes at frequencies below cutoff. This version of TABLE should replace the version in Appendix A when a tabulation of R_{11} and R_{12} is desired.

```
      SUBROUTINE TABLE(OM1,OM2,V1,V2,NOM,NVP,THETK,OM,V,INMODE,NOPT)
C     TABLE (SUBROUTINE)                      7/19/68   LAST CARD IN DECK IS NO.
C
C
C                  ----ABSTRACT----
C
C TITLE - TABLE
C          GENERATION OF SUSPICIONLESS TABLE OF NORMAL MODE DISPERSION
C          FUNCTION SIGNS
C
C                  TABLE CALLS SUBROUTINE MPOUT TO CONSTRUCT THE MATRIX OF
C                  NORMAL MODE DISPERSION FUNCTION SIGNS INMODE (STORED IN
C                  VECTOR FORM COLUMN AFTER COLUMN) FOR REGION IN FREQUENCY-
C                  PHASE VELOCITY PLANE (OM1.LE.OMEGA.LE.OM2.AND.V1.LE.VP.LE
C                  .V2).  SUBROUTINE SUSPCT IS CALLED TO EVALUATE THE SUSPI-
C                  CION INDEX ,ISUS, OF EACH INTERIOR ELEMENT IN THE MATRIX
C                  SCANNING FROM LEFT TO RIGHT, TOP TO BOTTOM. IF ISUS .NE.
C                  0 , INMODE IS ALTERED AS FOLLOWS.
C                        ISUS=1 ROW ADDED ABOVE SUSPICIOUS ELEMENT AND COLUMN
C                               ADDED TO ITS LEFT
C                            =2 COLUMN ADDED TO RIGHT OF SUSPICIOUS ELEMENT
C                               AND ROW ADDED ABOVE IT
C                            =3 ROW ADDED BELOW SUSPICIOUS ELEMENT AND COLUMN
C                               ADDED TO ITS RIGHT
C                            =4 COLUMN ADDED TO LEFT OF SUSPICIOUS ELEMENT
C                               AND ROW ADDED BELOW IT
C                  HOWEVER, NEITHER THE NUMBER OF ROWS NVP NOR THE NUMBER OF
C                  COLUMNS NOM WILL BE INCREASED BEYOND 100.  IF ISUS CALLS
C                  FOR AN ADDITIONAL ROW WHEN NVP = 100  , THE MESSAGE
C                  (NVP = 100            N = XX      M = XX) WILL BE PRINTED.
C                  N IS ROW NO. OF SUSPICIOUS ELEMENT.  M IS COLUMN NO.  IF
C                  ISUS CALLS FOR ADDITION OF A COLUMN WHEN NOM = 100, THE
C                  MESSAGE (NOM = 100            N = XX      M = XX) IS PRINTED.
C                  WHEN INMODE HAS BEEN EXPANDED SCANNING IS RESUMED AT THE
C                  ELEMENT IN NEW MATRIX WITH SAME ROW AND COLUMN NOS. AS
C                  THOSE OF SUSPICIOUS ELEMENT IN OLD MATRIX.  IF NOPT IS
C                  POSITIVE INMODE WILL BE PRINTED AS IT IS RETURNED FROM
C                  MPOUT AND IN ITS FINAL FORM.
C
C LANGUAGE    - FORTRAN IV (360, REFERENCE MANUAL - C28-6515-4)
C
C AUTHOR      - J.W.POSEY, M.I.T., JUNE,1968
C
C
C                        ----USAGE----
C
C     SUBROUTINES MPOUT,SUSPCT,LNGTHN,WIDEN,NMDFN ARE CALLED IN TABLE.
C
C FORTRAN USAGE
C     CALL TABLE(OM1,OM2,V1,V2,NOM,NVP,THETK,OM,V,INMODE,NOPT)
C
C INPUTS
C
C      OM1         MINIMUM VALUE OF FREQUENCY TO BE CONSIDERED.
C        R*4
C      OM2         MAXIMUM VALUE OF FREQUENY TO BE CONSIDERED
C        R*4
```

```
      . . . . .
C     V1         MINIMUM VALUE OF PHASE VELOCITY TO BE CONSIDERED
C       R*4
C     V2         MAXIMUM VALUE OF PHASE VELOCITY TO BE CONSIDERED
C       R*4
C     NOM        INITIAL NO. OF FREQUENCIES TO BE CONSIDERED
C       I*4
C     NVP        INITIAL NO. OF PHASE VELOCITIES TO BE CONSIDERED
C       I*4
C     THETK      PHASE VELOCITY DIRECTION (RADIANS)
C       R*4
C     NOPT       PRINT OUT OPTION.  IF NOPT = -1, NO PRINT.  IF NOPT = 1,
C       I*4      INMODE IS PRINTED IN ITS INITIAL FORM (GENERATED BY MPOUT)
C                AND IN ITS FINAL FORM.
C
C OUTPUTS
C
C     NOM        TOTAL NO. OF FREQUENCIES CONSIDERED
C       I*4
C     NVP        TOTAL NO. OF PHASE VELOCITIES CONSIDERED
C       I*4
C     OM         VECTOR WHOSE ELEMENTS ARE THE VALUES OF ANGULAR FREQUENCY
C       R*4(D)   CORRESPONDING TO THE COLUMNS OF THE INMODE MATRIX
C
C     V          VECTOR WHOSE ELEMENTS ARE THE VALUES OF PHASE VELOCITY
C       R*4(D)   CORRESPONDING TO THE ROWS OF THE INMODE MATRIX
C
C     INMODE     EACH ELEMENT OF THIS MATRIX CORRESPONDS TO A POINT IN THE
C       I*4(D)   FREQUENCY (OM) - PHASE VELOCITY (V) PLANE.  IF THE NORMAL
C                MODE DISPERSION FUNCTION (FPP) IS POSITIVE AT THAT POINT,
C                THE ELEMENT IS +1, IF FPP IS NEGATIVE, THE ELEMENT IS -1,
C                IF FPP DOES NOT EXIST, THE ELEMENT IS 5.  INMODE HAS NVP
C                ROWS AND NOM COLUMNS.  MATRIX IS STORED AS A VECTOR,
C                COLUMN AFTER COLUMN.
C
C
C                      ----EXAMPLE----
C
C LET INMODE = -1,5,5,5,1,-1,-1,-1,1,1,-1,-1,1,1,1,1
C     WITH NOM = NVP = 4
C     AND OM = 1.0,1.5,2.0,2.5            THETK = 3.14159
C         V = 1.0,2.0,3.0,4.0
C (VALUES NOT CORRECT, FOR ILLUSTRATION ONLY)
C
C THEN THE TABLE WILL BE PRINTED AS FOLLOWS.
C
C VPHASE       NORMAL MODE DISPERSION FUNCTION SIGN
C  1.00000   -+++
C  2.00000   X-++
C  3.00000   X--+
C  4.00000   X--+
C
C       OMEGA 1234
C              PHASE VELOCITY DIRECTION IS   90.0000EGREES
C
C OMEGA =
C     0.10000E 01   0.15000E 01   0.20000E 01   0.25000E 01
C
```

```
C
C                    ----PROGRAM FOLLOWS BELOW----
C
C
C
      DIMENSION OM(100),V(100),INMODE(10000),DORN(100),KORN(100)
      DIMENSION RPP(2,2)
      COMMON IMAX,CI(100),VXI(100),VYI(100),HI(100)
C
C MPOUT IS CALLED TO PRODUCE INMODE MATRIX AND OM AND V VECTORS.
      CALL MPOUT(OM1,OM2,V1,V2,NOM,NVP,INMODE,OM,V,THETK)
C
C IFLAG = 1 INDICATES FIRST TIME THROUGH WRITE PROCEDURE
      IFLAG = 1
C
C INMODE IS PRINTED IF NOPT IS POSITIVE
      IF (NOPT.GE.0) GO TO 123
    5 IFLAG = 0
      NOPER=0
C NOPER IS THE NUMBER OF EXPANSION OPERATIONS PERFORMED IN THE PRESENT
C SCAN OF THE MATRIX.  THUS, NOPER IS THE NUMBER OF SUSPICIOUS POINTS
C FOUND IN THE PRESENT SCAN.
C
C BEGIN SCANNING OF INTERIOR ELEMENTS OF INMODE IN UPPER LEFT CORNER
      N = 2
      M = 2
   10 CALL SUSPCT(N,M,NVP,INMODE,ISUS)
C
C POINT (N,M) IS SUSPICIOUS IF ISUS.NE.0
      IF(ISUS.NE.0) GO TO 60
C
C CHECK FOR END OF ROW
   20 IF (M.LT.(NOM-1)) GO TO 30
C
C CHECK FOR LAST ROW
      IF (N.LT.(NVP-1)) GO TO 40
      GO TO 121
C
C MOVE ONE COLUMN TO RIGHT
   30 M = M+1
      GO TO 10
C
C ADVANCE ONE ROW AND START AT COLUMN TWO
   40 N = N+1
      M = 2
      GO TO 10
C
C CHECK FOR MAXIMUM VALUE OF NVP
   60 IF(NVP.LT.100) GO TO 62
   61 FORMAT (24H NVP = 100           N =,I3,8H     M =,I3)
      WRITE (6,61) N,M
      GO TO 20
   62 IF(NOM .LT. 100) GO TO 70
   63 FORMAT(24HNOM = 100            N=,I3, 8H      M=,I3)
   64 WRITE(6,63) N,M
      GO TO 20
   70 IF(ISUS .NE. 1) GO TO 75
```

```
C
C  ADD ROW ABOVE SUSPICIOUS POINT
      N1=N-1
C
C  ADD A COLUMN TO LEFT OF SUSPICIOUS POINT
      M1=M-1
      GO TO 100
   75 IF(ISUS .NE. 2) GO TO 80
C
C  ADD A COLUMN TO RIGHT OF SUSPICIOUS POINT
      M1=M
C
C  ADD ROW ABOVE SUSPICIOUS POINT
      N1=N-1
      GO TO 100
   80 IF(ISUS .NE. 3) GO TO 85
C
C  ADD A COLUMN TO RIGHT OF SUSPICIOUS POINT
      M1=M
C
C  ADD ROW BELOW SUSPICIOUS POINT
      N1=N
      GO TO 100
C
C  ADD ROW BELOW SUSPICIOUS POINT
   85 N1=N
C
C  ADD A COLUMN TO LEFT OF SUSPICIOUS POINT
      M1=M-1
  100 CONTINUE
      CALL LNGTHN(OM,V,INMODE,NOM,NVP,NVPP,N1,1,THETK)
      CALL WIDEN(OM,V,INMODE,NOM,NOMP,NVPP,M1,1,THETK)
      NVP=NVPP
      NOM=NOMP
      NOPER=NOPER+1
      GO TO 10
  121 CONTINUE
      IF(NOPER .GT. 0 .AND. NVP .LT. 100 .AND. NOM .LT. 100) GO TO 5
C
C DO NOT PRINT INMODE IF NOPT IS NEGATIVE
      IF(NOPT .LT. 0) RETURN
C
C LABELING
  122 FORMAT (6H1VPHSE,6X,36HNORMAL MODE DISPERSION FUNCTION SIGN/)
  123 WRITE (6,122)
      DO 133 I=1,NVP
      DO 128 J=1,NOM
      J88=(J-1)*NVP+I
      J89=INMODE(J88)-1
      IF (J89) 126,125,124
  124 CONTINUE
C
C IF INMODE = 5, DORN = 1HX
      DATA Q1/1HX/
      DORN(J) = Q1
      GO TO 127
  125 CONTINUE
```

```
C
C IF INMODE = 1, DORN = 1H+
      DATA Q2/1H+/
      DORN(J) = Q2
      GO TO 127
  126 CONTINUE
C
C IF INMODE = -1, DORN = 1H-
      DATA Q3/1H-/
      DORN(J) = Q3
  127 CONTINUE
  128 CONTINUE
C
C PRINT ROW I OF TABLE
      WRITE (6,130)V(I),(DORN(J), J=1,NOM)
  130 FORMAT(1H ,F8.5,3X,100A1)
  133 CONTINUE.
      J10 = 10
      DO 150 J=1,NOM
C
C NUMBER COLUMNS
  150 KORN(J) = MOD(J,J10)
      WRITE (6,213) (KORN(J), J=1,NOM)
  213 FORMAT (6HOOMEGA,6X,100I1)
C
C CONVERT THETK FROM RADIANS TO DEGREES
      X = THETK*180/3.14159
      WRITE (6,413) X
  413 FORMAT (1H ,11X,27HPHASE VELOCITY DIRECTION IS,F9.3,
     1   8HDEGREES )
      WRITE (6,513)
  513 FORMAT ( 8HOOMEGA =)
C
C LIST VALUES OF OMEGA WHICH CORRESPOND TO COLUMNS OF TABLE
      WRITE (6,613) (OM(I),I=1,NOM)
  613 FORMAT ( 1H ,5E14.5)
C
C IF SUSPICION ELIMINATION HAS NOT BEEN PERFORMED, BEGIN IT AT THIS TIME
      IF(IFLAG.EQ.1) GO TO 5
      DELVP=(V2-V1)/(NVP-1)
      OMEGK=OM1
      DELOM=(OM2-OM1)/(NOM-1)
      DO 988 IAA=1,NOM
      WRITE (6,933) OMEGK
 933  FORMAT (1H ,3X,6HOMEGA=,E14.5)
      DO 977 JAA=1,NVP
      VE=V1+(JAA-1)*DELVP
      AKX=OMEGK/VE
      AKY=0.0
      CALL RRRR(OMEGK,AKX,AKY,RPP,KY)
      WRITE (6,944) VE,RPP(1,1),RPP(1,2)
 944  FORMAT (1H ,E12.5,6X,E12.5,3X,E12.5)
 977  CONTINUE
      OMEGK=OMEGK+DELOM
 988  CONTINUE
      RETURN
      END
```